Tampere University

IBRAHIM TOUMAN

IF-LEVEL SIGNAL-PROCESSING OF GPS AND GALILEO RADIONAVIGATION SIGNALS USING MATLAB/SIMULINK®

Including Effects of Interference and Multipath

Faculty of Computing and
Electrical Engineering
Master of Science Thesis
December 2019

ABSTRACT

Ibrahim Touman: IF-level Signal-Processing of GPS and Galileo Radionavigation Signals Using MATLAB/Simulink®: Including Effects of Interference and Multipath
Master of Science Thesis
Tampere University
Master's Degree Programme in Electrical Engineering
December 2019

Open-source GNSS simulator models are rare and somewhat difficult to find. Therefore, Laboratory of Electronics and Communications Engineering in the former Tampere University of Technology (and now Tampere University, Hervanta Campus) has took it upon itself to develop, from time to time, a free and open-source simulator model based on MATLAB/Simulink® for signal processing of a carefully selected set of GNSS radionavigation signals, namely, Galileo E1, Galileo E5, GPS L1, and GPS L5. This M.Sc. thesis is the culmination of those years which have been spent intermittently on research and development of that simulator model. The first half of this M.Sc. thesis is a literature review of some topics which are believed to be of relevance to the thesis's second half which is in turn more closely associated with documenting the simulator model in question. In particular, the literature review part presents the reader with a plethora of GNSS topics ranging from history of GNSS technology to characteristics of existing radionavigation signals and, last but not least, compatibility and interoperability issues among existing GNSS constellations. While referring to the GNSS theory whenever necessary, the second half is, however, mainly focused on describing the inner-workings of the simulator model from the standpoint of software implementations. Finally, the second half, and thereby the thesis, is concluded with a presentation of various statistical results concerning signal acquisition's probabilities of detection and false-alarm, in addition to signal tracking's RMSE.

Keywords: GNSS, GPS, Galileo, RFC, Interoperability, Multipath, Interference, Signal Acquisition, Probability of Detection, Probability of False Alarm, Tracking RMSE

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND SYMBOLS

TUT	Tampere University of Technology
TAU	Tampere University
M.Sc.	Master of Science
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GLONASS	GLObal'naya NAvigatsionnaya Sputnikovaya Sistema
BDS	BeiDou navigation satellite System
IF	Intermediate Frequency
GUI	Graphical User-Interface
GUIDE	Graphical User-Interface Development Environment
CLI	Command-Line Interface
RF	Radio Frequency
RFC	Radio-Frequency Compatibility
RFI	Radio-Frequency Interference
RMSE	Root-Mean-Square Error
SPS	Standard Positioning Service
PPS	Precision Positioning Service
SA	Selective Availability
EU	European Union
US	United States
TOT	time-of-transmission
TOA	time-of-arrival
TDOA	time-difference-of-arrival
UERE	User Equivalent Range Error
DOP	Dilution of Precision
GDOP	Geometric Dilution of Precision
SNR	Signal to Noise power Ratio
PVT	Position, Velocity, and Time
PRN	Pseudo-Random Noise
A/D	Analog-to-Digital-conversion
NRZ	Non-Return to Zero
DSSS	Direct-Sequence Spread-Spectrum
BPSK	Binary Phase-Shift Keying
BPSK-R	Binary Phase-Shift Keying with Rectangular pulse-shape
QPSK	Quadrature Phase-Shift Keying
WSCS	Wide-Sense Cyclo-Stationary
LFSR	Linear-Feedback Shift Register
C/A	Coarse/Acquisition
CCF	Cross-Correlation Function
ACF	Auto-Correlation Function
PSD	Power Spectral Density
PAM	Pulse-Amplitude Modulation
SRRC	Square-Root Raised-Cosine
BOC	Binary Offset Carrier
CBOC	Composite Binary Offset Carrier
MBOC	Multiplexed Binary Offset Carrier
AltBOC	Alternative Binary Offset Carrier
I/Q	In-phase/Quadrature
FDMA	Frequency-Division Multiple Access
TDMA	Time-Division Multiple Access
CDMA	Code-Division Multiple Access
DS-CDMA	Direct-Sequence – Code-Division Multiple Access

SDMA	Space-Division Multiple Access
OFDM	Orthogonal Frequency-Division Multiplexing
IEEE	Institute of Electrical and Electronics Engineers
MAI	Multiple-Access Interference
MAI-A	Multiple-Access Interference – intra-system
MAI-R	Multiple-Access Interference – inter-system
MAI-A/R	Multiple-Access Interference – intra-system and/or inter-system
N/A	Not Applicable, Not Available, or No Answer
PRS	Public Regulated Service
OS	Open Service
SoL	Safety of Life
CS	Commercial Service
FEC	Forward Error Correction
N-H	Neumann-Huffman
AFSPC	Air Force Space Command
USAF	United States Air Force
DAF	Department of the Air Force
DoD	Department of Defense
CNSA	China National Space Administration
GSA	European GNSS Agency
SSC	Spectral Separation Coefficient
CT_SSC	Code Tracking - Spectral Sensitivity Coefficient
ITU	International Telecommunication Union
ICG	International Committee on GNSS
TRF	Terrestrial Reference Frame
ECEF	Earth-Centered and Earth-Fixed
ITRF	International Terrestrial Reference Frame
LPF	Low-Pass Filter
BPF	Band-Pass Filter
MEO	Medium Earth Orbit
MDR	Multipath-to-Direct Ratio
PDP	Power-Delay Profile
FIR	Finite Impulse Response
CIR	Channel Impulse Response
AWGN	Additive White Gaussian Noise
PDF	Probability Density Function
CWI	Continuous Wave Interference
SIR	Signal to Interference power Ratio
DME	Distance Measuring Equipment
MAGR	Miniature Airborne GPS Receiver
FFT	Fast Fourier Transform
IFFT	Inverse Fast Fourier Transform
SVN	Satellite Vehicle Numbers
NCO	Numerically Controlled Oscillator
PIT	Predetection Integration Time
PLL	Phase-Locked Loop
FLL	Frequency-Locked Loop
DLL	Delay-Locked Loop
EML	Early Minus Late
HRC	High Resolution Correlation
MGD	Multiple Gate Delay
RMSE	Root-Mean-Square Error
CPU	Central Processing Unit
PCHIP	Piecewise Cubic Hermite Interpolating Polynomial

c	speed of light in vacuum.
t_n^{TOT}	n^{th} satellite's time of transmission (time-stamped using satellite's clock).
t_n^{TOA}	n^{th} satellite's time of arrival (time-stamped using receiver's clock).
T_u	time offset between the clocks of space-segment and user-segment (attributed to asynchronicity).
ρ	pseudorange.
x_n	n^{th} satellite's x-coordinate.
y_n	n^{th} satellite's y-coordinate.
z_n	n^{th} satellite's z-coordinate.
x_u	user-segment's x-coordinate.
y_u	user-segment's y-coordinate.
z_u	user-segment's z-coordinate.
k	positive integer which is used for sample-indexing.
$a[k]$	multi-level (and sometimes complex-valued) symbol sequence.
$\tilde{a}[k]$	logical-bit sequence.
T_c	chip-period.
R_c	chipping-rate.
R_b	bit-rate.
T_{PRN}	repetition-period of PRN spreading code.
$c[k]$	arbitrary discrete-time PRN spreading code chip sequence.
$d[k]$	arbitrary discrete-time data symbol sequence.
$x[k]$	discrete-time DSSS chip sequence.
$g(t)$	arbitrary real-valued and continuous-time pulse-shape filter.
$\delta(t)$	continuous-time Dirac delta function.
$x(t)$	continuous-time baseband PAM waveform comprising DSSS chips.
$U(t)$	continuous-time Heaviside unit-step function.
$g_{\sqcap}(t)$	rectangular pulse-shape.

$R_{\sqcap}(\tau)$	ACF of $g_{\sqcap}(t)$.
$G_{\sqcap}(f)$	PSD of $g_{\sqcap}(t)$.
T_{sc}	period of a sine or cosine square wave subcarrier.
f_{sc}	frequency of a sine or cosine square wave subcarrier.
f_r	reference frequency.
n_{sc}	an integer multiplier of f_r within f_{sc}.
$g_{n_{sc}}(t)$	a sine or cosine square wave subcarrier waveform.
n_c	integer multiplier of f_r within R_c.
$g_{n_c}(t)$	a rectangular pulse-shape.
$g_{\mathrm{BOC}(n_{sc},\, n_c)}(t)$	a $\mathrm{BOC}(n_{sc}, n_c)$ pulse-shape.
$G_{\mathrm{BOC}(n_{sc},\, n_c)}(f)$	PSD of a sine-phased $g_{\mathrm{BOC}(n_{sc},\, n_c)}(t)$.
f_0	fundamental frequency.
T_{sim}	the time duration it takes the underlying computer to complete a given simulation.
f_c	carrier frequency.
$v_{\mathrm{BP}}(t)$	arbitrary bandpass signal.
$a(t)$	instantaneous amplitude of $v_{\mathrm{BP}}(t)$.
$\phi(t)$	instantaneous phase of $v_{\mathrm{BP}}(t)$.
$v_{\mathrm{I}}(t)$	in-phase component of $v_{\mathrm{BP}}(t)$.
$v_{\mathrm{Q}}(t)$	quadrature component of $v_{\mathrm{BP}}(t)$.
$\boldsymbol{v}_{\mathrm{LP}}(t)$	complex-valued lowpass-equivalent of $v_{\mathrm{BP}}(t)$.
f_{IF}	intermediate frequency.
R_s	sampling-rate.
T_s	fundamental sampling-period.
$\boldsymbol{v}_{\mathrm{LP}}(kT_s)$	an arbitrary discrete-time (and usually complex-valued) lowpass-equivalent signal.
$\boldsymbol{v}_{\mathrm{IF}}(kT_s)$	an IF-centered analytic version of $\boldsymbol{v}_{\mathrm{LP}}(kT_s)$.
θ_{L1}	nominal (but ambiguous) carrier phase-shift of GPS L1 signal.

f_{L1}	carrier frequency for GPS L1 signal.
$g^{\mathrm{C/A}}(t)$	baseband PAM waveform of the PRN spreading code of GPS C/A channel.
$g^{\mathrm{P(Y)}}(t)$	baseband PAM waveform of the PRN spreading code of GPS P(Y) channel.
$d^{\mathrm{L1}}(t)$	baseband PAM waveform of GPS signal's navigation data.
$P_{\mathrm{C/A}}$	power coefficient for the GPS L1 C/A channel.
$P_{\mathrm{P(Y)}}$	power coefficient for the GPS L1 P(Y) channel.
$x_{\mathrm{L1}}(t)$	(BPSK-R)-modulated GPS L1 bandpass signal.
$x_{\mathrm{LP}}^{\mathrm{C/A}}(t)$	low-pass equivalent GPS L1 C/A channel extracted from $x_{\mathrm{L1}}(t)$.
$\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{C/A}}(kT_s)$	a discrete-time, complex-valued, and IF-centered GPS L1 C/A analytic TX-signal.
θ_{L5}	nominal (but ambiguous) carrier phase-shift of GPS L5 signal.
f_{L5}	carrier frequency for GPS L5 signal.
$g^{\mathrm{L5\text{-}I}}(t)$	baseband PAM waveform of the primary PRN Spreading code for the GPS L5-I channel.
$g^{\mathrm{L5\text{-}Q}}(t)$	baseband PAM waveform of the primary PRN Spreading code for the GPS L5-Q channel.
$h^{\mathrm{L5\text{-}I}}(t)$	baseband PAM waveform of the GPS L5-I channel's length 10 Neumann–Hoffman secondary code.
$h^{\mathrm{L5\text{-}Q}}(t)$	baseband PAM waveform of the GPS L5-Q channel's length 20 Neumann–Hoffman secondary code.
$d^{\mathrm{L5\text{-}I}}(t)$	baseband PAM waveform of the GPS L5-I channel's navigation data.
$P_{\mathrm{L5\text{-}I}}$	power coefficient for the GPS L5-I channel.
$P_{\mathrm{L5\text{-}Q}}$	power coefficient for the GPS L5-Q channel.
$x_{\mathrm{L5}}(t)$	QPSK-modulated GPS L5-I/Q composite bandpass signal.
$\boldsymbol{x}_{\mathrm{LP}}^{\mathrm{L5}}(t)$	low-pass equivalent version of $x_{\mathrm{L5}}(t)$.
$\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{L5}}(kT_s)$	a discrete-time, complex-valued, and IF-centered GPS L5-I/Q composite analytic TX-signal.
$\hat{r}$	a unit vector pointing from the user-segment towards the satellite in question.

$\overrightarrow{\Delta v}$	a satellite to user-segment relative velocity vector.
f_{D}	Doppler-shift frequency with respect to f_c.
$\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$	Doppler-shifted version of $\boldsymbol{x}_{\mathrm{IF}}(kT_s)$.
N_{mp}	total number of multipath components (or taps), including the line-of-sight component.
$\boldsymbol{y}_{\mathrm{BP}}(t)$	continuous-time, complex-valued, analytic, and bandpass post-multipath received signal.
α_0	line-of-sight component's received magnitude.
α_n	n^{th} reflected multipath component's received magnitude.
$\widetilde{\alpha_n}$	Multipath-to-Direct-magnitude-Ratio (MDR).
τ_0	line-of-sight component's propagation delay.
τ_n	n^{th} reflected multipath component's propagation delay.
$\widetilde{\tau_n}$	multipath excess delay.
θ_0	nominal carrier phase-shift.
$\widetilde{f_n}$	n^{th} reflected multipath component's received carrier frequency relative to the direct component's received carrier frequency.
φ_0	line-of-sight component's received carrier phase-shift.
φ_n	n^{th} reflected multipath component's received carrier phase-shift.
b_n	n^{th} multipath component's propagation delay in chip.
M_n	n^{th} multipath component's propagation delay in sample.
P_x	average power level of the pre-multipath signal $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$.
$\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$	multipath-distorted version of $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$.
P_y	average power level of the post-multipath signal $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$.
α	scaling coefficient which ensures that $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$ has the same average power as of $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$.
$h_{\mathrm{CIR}}[k]$	time-invariant and discrete-time Channel Impulse Response (CIR).
$\delta[k]$	discrete-time Dirac delta function.
$G_{\mathfrak{n}}(f)$	PSD of thermal noise.
N_0	noise spectral density.

K_{B}	Boltzmann constant.
$\mathcal{T}$	system's noise temperature in kelvin.
C/N_0	carrier-to-noise-density-power-ratio.
$H_{\mathrm{BPF}}(f)$	transfer function of a real-valued, unity gain, analog, and ideal Band-Pass Filter (BPF).
f_{BPF}	center frequency of the BPF.
B_{BPF}	single-sided bandwidth of the BPF.
$\mathfrak{n}_{\mathrm{BP}}(t)$	bandpass noise (i.e., thermal noise filtered by BPF).
$G_{\mathfrak{n}}^{\mathrm{BP}}(f)$	PSD of the bandpass noise.
$P_{\mathfrak{n}}$	average power level of the bandpass noise.
$\sigma_{\mathfrak{n}}^2$	variance of the bandpass noise.
$\mu_{\mathfrak{n}}$	mean of the bandpass noise.
$\mathbf{n}_{\mathrm{LP}}(t)$	lowpass-equivalent of bandpass noise.
$\mathbf{n}_{\mathrm{LP}}(kT_s)$	discrete-time version of $\mathbf{n}_{\mathrm{LP}}(t)$.
$\mathfrak{n}_{\mathrm{I}}(t)$	in-phase component of $\mathbf{n}_{\mathrm{LP}}(t)$.
$\mathfrak{n}_{\mathrm{I}}(kT_s)$	discrete-time version of $\mathfrak{n}_{\mathrm{I}}(t)$.
$\mathfrak{n}_{\mathrm{Q}}(t)$	quadrature component of $\mathbf{n}_{\mathrm{LP}}(t)$.
$\mathfrak{n}_{\mathrm{I}}(kT_s)$	discrete-time version of $\mathfrak{n}_{\mathrm{Q}}(t)$.
$\mathcal{N}(\mu_{\mathfrak{n}},\ \sigma_{\mathfrak{n}}^2)$	Probability Density Function (PDF) of a normal distribution whose mean is $\mu_{\mathfrak{n}}$ and whose variance is $\sigma_{\mathfrak{n}}^2$.
$\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$	noise-distorted version of $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$.
P_z	average power level of the pre-CWI signal $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$.
$\mathfrak{i}\,(kT_s)$	discrete-time Continuous Wave Interference (CWI) signal.
N_{CWI}	total number of different frequency-shifted and amplitude-scaled sinusoidal components comprising the CWI signal $\mathfrak{i}\,(kT_s)$.
f_n	frequency offset for the n^{th} CWI component.
β_n	scaling coefficient for the n^{th} CWI component.
f_{CWI}	common frequency offset for all CWI components.

$\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$	(wideband interference)-distorted and / or CWI-distorted version of $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$.
$\hat{\mathrm{i}}\,(kT_s)$	estimated narrowband interference discrete-time signal.
$\boldsymbol{q}_{\mathrm{IF+D}}(kT_s)$	(narrowband interference)-mitigated version of $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$.
$\widehat{f_{\mathrm{D}}}$	coarse estimate of f_{D}.
τ_c	code-delay.
$\widehat{\tau_c}$	coarse estimate of τ_c.
M_0	number of samples by which the line-of-sight component is delayed.
M_c	number of samples by which the PRN spreading code is delayed.
$\widehat{M_c}$	estimate of M_c.
M_{PRN}	total number of samples comprising a T_{PRN}.
f_m	discretized frequency axis of the acquisition search space.
τ_ℓ	discretized delay axis of the acquisition search space.
f_{limit}	uncertainty range over which f_{D} is searched equally in opposite directions relative to f_{IF} in the acquisition search space.
Δf_m	frequency spacing (aka, frequency bin size) between any two successive frequency steps of f_m.
ξ	dimensionless coefficient whose value does not exceed 1.
$L_{\max}(\xi)$	maximum power loss in correlations' power levels due to resolution of the frequency axis of the search space, which is dependent on ξ.
Δf_m	frequency spacing (aka, frequency bin size) between any two successive frequency steps of f_m.
M_ℓ	the finite number of uniformly spaced delay steps in samples (aka, delay bins in samples) over which the number of samples by which the PRN spreading code was delayed is searched.
ΔM_ℓ	delay spacing in sample (aka, delay bin size in sample) between any two successive delay steps of M_ℓ.
Δb	delay bin size in chip.
$\Delta\tau_\ell$	delay bin size in time.
T_{coh}	coherent integration duration in signal acquisition.
N_{coh}	integer multiplier of T_{PRN} within T_{coh}.

T_{ncoh}	non-coherent integration duration (aka, dwell time) in signal acquisition.
N_{ncoh}	integer multiplier of T_{coh} within T_{ncoh}.
v	integer index used for numbering search spaces pairs.
$\boldsymbol{S}_v^{\text{data}}[f_m, \ell\ ]$	data channel coherent search spaces of the v^{th} pair.
$\boldsymbol{S}_v^{\text{pilot}}[f_m, \ell\ ]$	pilot channel coherent search spaces of the v^{th} pair.
$S_v^{\text{ncoh}}[f_m, \ell]$	noncoherent combined search space of the v^{th} pair.
$S_{\text{ncoh}}[f_m, \ell]$	the overall noncoherent search space.
$c_{\text{data}}(\ell T_s)$	local replica of the data channel.
$\boldsymbol{C}_{\text{data}}[n]$	Discrete Fourier Transform (DFT) with respect to the domain of delay bins in samples ℓ of $c_{\text{data}}(\ell T_s)$.
$c_{\text{pilot}}(\ell T_s)$	local replica of the pilot channel.
$\boldsymbol{C}_{\text{pilot}}[n]$	DFT with respect to ℓ of $c_{\text{pilot}}(\ell T_s)$.
$\boldsymbol{Q}_v[f_m, n]$	DFT with respect to ℓ of a version of $\boldsymbol{q}_{\text{IF+D}}(\ell T_s)$ which is frequency-shifted by $(f_{\text{IF}} + f_m)$.
f_{s}	a number that determines the range of frequencies which are suppressed, if possible, equally in opposite directions relative to the frequency coordinate of the global peak in $S_{\text{ncoh}}[f_m, \ell]$.
f_m^{supp}	a version of f_m which excludes the range of frequencies which ought to be suppressed in $S_{\text{ncoh}}[f_m, \ell]$.
M_{s}	the number of samples which are suppressed equally in opposite directions relative to the delay coordinate in sample of the global peak in $S_{\text{ncoh}}[f_m, \ell]$.
b_{s}	the number of chips which are suppressed equally in opposite directions relative to the delay coordinate in chip of the global peak in $S_{\text{ncoh}}[f_m, \ell]$.
ℓ_{supp}	a version of ℓ which excludes the range of delays which ought to be suppressed in $S_{\text{ncoh}}[f_m, \ell]$.
$S_{\text{supp}}[f_m, \ell]$	a version of $S_{\text{ncoh}}[f_m, \ell]$ in which the global peak and its vicinity have been suppressed (i.e., set to zero).
γ	a threshold whose value is compared against the ratio between the global peak and the second highest peak in $S_{\text{ncoh}}[f_m, \ell]$.
$\Delta_{\text{E-L}}$	double the phase-spacing between any two successive replicas found in the tracking module's bank of correlators.

ω_{0f}	natural radian frequency of a Frequency-Locked Loop (FLL).
B_{nf}	frequency noise bandwidth of the carrier loop filter.
ω_{0p}	natural radian frequency of a Phase-Locked Loop (PLL).
B_{np}	phase noise bandwidth of the carrier loop filter.
ω_{DLL}	natural radian frequency of the Delay-Locked Loop (DLL).
B_{DLL}	frequency noise bandwidth of the code loop filter.
N_{reg}	the number of bits comprising the holding registers in the NCOs of both code and carrier tracking loops in signal tracking modules.
N_{bank}	length of the list of phases in the bank of correlators found in signal tracking modules.
p_{d}	probability of detection in signal acquisition.
p_{fa}	probability of false-alarm in signal acquisition.
H_0	null hypothesis.
H_1	alternative hypothesis.
M_{ε}	maximum tolerated error in the code-delay estimate in sample unit.
b_{ε}	maximum tolerated error in the code-delay estimate in chip unit.
$\mathcal{A}$	arbitrary random event.
$P(\mathcal{A})$	probability of event $\mathcal{A}$.
N_{exp}	total number of performed experiments (or trials).
$n_{\mathcal{A}}$	total number of times event $\mathcal{A}$ occurred while performing experiments.
σ	true delay against which estimated delay values are compared.
$\widehat{\sigma_n}$	n^{th} estimated delay value.
N_{σ}	total number of estimated delay values $\widehat{\sigma_n}$ over which the tracking RMSE is evaluated.

1. INTRODUCTION

1.1 Background

Although satellite positioning services for civilian users were rather sluggish when they were offered for the first time in the 1980s by the GPS constellation, they have, nevertheless, picked up the pace in the few ensuing decades to become an integral part of modern-day civilization. In fact, GNSS technology has become, one way or another, the de facto technique for positioning and navigation nowadays. GNSS technology proved itself to be a very reliable and convenient technology.

In the last decade or so, new GNSS radionavigation signals with various degrees of complexity have become operational. The compatibility of those new signals with the older and well-established signals is still not fully investigated, and hence the need arises for free and open-source GNSS simulators which incorporate latest radionavigation signals' specifications. To the best of the author knowledge, those free and open-source GNSS simulators are by no means numerous. Therefore, this M.Sc. thesis tries to address this shortage by increasing the number of those few free and open-source GNSS simulators by one. In particular, a complete simulator for IF-level signal processing of Galileo E1, Galileo E5, GPS L1, and GPS L5 radionavigation signals, which is based on MATLAB/Simulink®, was developed alongside this M.Sc. thesis.

1.2 State of the Art

One of the new trends in GNSS technology is to use multi-constellation and/or dual-frequency positioning. The first known commercial prototype of a GNSS receiver capable of multi-constellation positioning is the one which was designed in the year 1990 by a company called Magnavox (for more details refer to *Section 4.3*). Due to the lack of interoperability between the only two available GNSS constellations back in 1990, namely GPS and GLONASS, it was a daunting task for Magnavox to design that receiving equipment [1]. However, although it was quite early for manufacturing a GNSS receiver capable of multi-constellation positioning in 1990; nevertheless, multi-constellation positioning

became the norm in mainstream GNSS receivers in the few ensuing decades. Nowadays, almost every high-end commercial smartphone is capable of multi-constellation positioning.

Due to cost, complexity, and relative novelty of open radionavigation signals on multiple disjoint bands, dual-frequency positioning turned out to be more challenging to implement than multi-constellation positioning, especially in mass-market positioning products. Therefore, the potentials of dual-frequency positioning were untapped by all available commercial smartphones, and this continued to be the case until very recently. To the best of the author's knowledge, the first time a dual-frequency GNSS chipset which is dedicated for upcoming smartphone generations is announced, was as recently as September 2017 by a company called Broadcom Limited [2]. It is worth mentioning that, Broadcom named their chipset BCM47755, and claimed it is capable of centimeter-level positioning accuracy with minimal power consumption and footprint. Moreover, BCM47755 was developed with the capacity to track all of the following signals at once: dual-frequency mode for both GPS L1/L5 and Galileo E1/E5a, as well as single-frequency mode for some of the signals belonging to remaining constellations (i.e., GLONASS and BDS) [3]. However, the first commercial smartphone that is embedded with BCM47755 was released in June 2018 under the brand name Mi 8 by Xiaomi Corporation. Interestingly, Xiaomi claimed that its new Mi 8 smartphone is capable of delivering positioning accuracy down to the decimeter-level, which is unprecedented in any earlier low-cost and mass-market product of any sort [4]. Eventually, within few years' time, smartphones capable of dual-frequency positioning are set to dominate the market of low-cost and handheld navigation devices [5].

1.3 Author's Contributions

- Elaborated an extensive literature review covering various GNSS topics, which include, among others, compatibility and interoperability issues among existing GNSS constellations.
- With the help of two separate and outdated simulator models for IF-level signal-processing of Galileo E1 and Galileo E5 radionavigation signals, which both were built using outdated versions of MATLAB/Simulink®, the author, however, managed to build a complete and coherent simulator model for IF-level signal-processing of any viable combination of active Galileo E1, Galileo E5, GPS L1, and

GPS L5 radionavigation signals using the latest version of 64-bit MATLAB/Simulink® that the author could get his hands on (namely, version 9.4.0.813654 (R2018a)).

- Built from scratch a *Graphical User-Interface* (*GUI*) dialog box which collects a handful of user-input parameters from the end-user using MATLAB's App Designer. Also, built from scratch another GUI panel which gives the end-user a visualization of various relevant end results using MATLAB's *Graphical User-Interface Development Environment* (*GUIDE*).
- Upgraded the simulator model with the capability of conducting statistical simulations while MATLAB/Simulink® is running inside a user-interface environment which supports *Command-Line Interface* (*CLI*) shells exclusively.
- Conducted tens of statistical simulations using a university-provided remotely controlled computer-cluster, which is based on Linux's CLI slurm workload manager.

1.4 Thesis Outline

In the second chapter, after explaining what is meant by GNSS technology and what important roles it plays in modern-day civilization, we strive to give a succinct overview to the reader about the civil use history of existing GNSS constellations. Also, the basic concept behind satellite navigation, which is range measurement, is briefly presented therein. Eventually, the error sources to which existing GNSS systems are susceptible are not left unmentioned.

In the third chapter, various technical characteristics of existing GNSS radionavigation signals are presented. Also, a tabular summary of the technical characteristics of the four radionavigation signals which are of interest to us in this M.Sc. thesis is given at the end of the chapter.

In the fourth chapter, we delve into some of the technicalities of Radio-Frequency Compatibility (RFC) and interoperability, both of which among existing GNSS constellations. The argument therein is rigourized by presenting some interesting simulation results of worst-case scenario Radio-Frequency Interference (RFI) taken from the literature concerned with RFC in GNSS. Furthermore, interoperability is thoroughly defined and its relation to RFC is presented. Also, the importance of interoperability is emphasized by

presenting an example of the challenges encountered while designing the first commercial prototype for a GNSS receiver capable of multi-constellation positioning back when interoperability was non-existent.

In the fifth chapter, a comprehensive overview is presented for the IF-level simulator model for studying Galileo E1, Galileo E5, GPS L1, and GPS L5 radionavigation signals using MATLAB/Simulink®, which was developed while elaborating this M.Sc. thesis. It is evident from the correlation between the title of the fifth chapter and the title of this M.Sc. thesis that the fifth chapter contains the core contents of this M.Sc. thesis.

In the sixth and final chapter, the difference between regular simulation mode and statistical simulation mode is explained thoroughly. Furthermore, probability of detection and probability of false-alarm, which both are properties of signal acquisition, are presented with accompanying Monte-Carlo simulations results. Finlay, the Root-Mean-Square Error (RMSE) of signal tracking is also covered with some accompanying statistical simulations' results.

2. GNSS TECHNOLOGY IN GENERAL

2.1 Introduction to GNSS

2.1.1 What is GNSS?

GNSS stands for *Global Navigation Satellite System*, which is a fairly complicated technology consisting of three segments, namely: space-segment, control-segment, and user-segment. The main purpose of GNSS is to provide a stationary or moving user with his/her precise 3D position (longitude, latitude, and altitude) anywhere on the surface of the earth, in near real-time scenario. However, the user benefiting from GNSS service is not necessarily a human being in the strict sense of the word, because for example, an autonomous machine (e.g. self-driving car) could navigate its way with the help of GNSS service as well. 3D positioning in GNSS is made possible due to the transmission of radionavigation signals by a constellation of artificial satellites. Consequently, any user who possesses the appropriate receiving equipment could lock onto those signals and extract his/her location anywhere on the surface of the earth with adequate level of accuracy in the process. Since the end-user does not need to exchange any sort of data back with the satellites from which he/she is receiving radio signals, GNSS technology is regarded a passive system from the standpoint of user-segment ([6] Ch. 8, pp. 299-303). In this context, passiveness should not be confused in terms of power consumption in receiving equipment; instead, it exclusively refers to the simplex (or uni-directional) nature of the transmission channel between any orbiting GNSS satellite and its corresponding receiving equipment [7].

It is more than likely that the human who first wandered the Earth was confronted by the navigation problem, where he/she realized the necessity of resorting to some sort or another of makeshift navigation technique in order to save himself/herself from aimless walking forever. Therefore, in an effort to solve the problem of finding someone's true location on the surface of the earth (i.e., positioning) or to figure out if the person is heading in the right direction leading him/her to the intended destination (i.e., navigation), many ingenious navigation techniques had been invented and deployed by pioneers and explorers throughout the past millenniums. However, with the emergence of a modern and sophisticated positioning and navigation technology such as GNSS, those old navi-

gation techniques have become obsolete, forgotten, or relegated to museums nowadays. Technology always wins; hence, the disruptive GNSS technology is the predominant method for positioning and navigation today, and it is superior to any preceding navigation technique to say the least ([8] Ch. 1, pp. 1-3). The innovative GNSS technology offers: round-the-clock reliable global coverage, free of direct-service fee, accurate positioning and timing, not to mention low-cost and small-sized handheld receiving equipment. These astounding features alone are worthy of making GNSS technology a formidable tool with remarkable benefits. Finally, it is not exaggeration when GNSS technology is dubbed as the Holy Grail for solving the vast majority of positioning and navigation problems, the very same problems which have baffled human beings since the onset of mankind.

2.1.2 Brief History of GNSS for Civil Use

NAVSTAR Global Positioning System (GPS) was developed primarily as a military technology, although this is disputed in [9]. One can hardly imagine that a technology such as GPS with its huge potentials for civilian applications will continue to be restricted as military technology forever. It is often stated that, the first official announcement promising the release of GPS service for civil use came in the year 1983 when GPS was still in experimental phase by Ronald Regan, who was the United States' president back then [10] ([11] Appen. B, pp. 247-248). However, the author in [12] argued that the first official announcement came two years prior to Regan's announcement. Regardless of that, Regan's promise gradually came to fruition over the course of several ensuing years, starting with the permission of GPS service for limited number of professional civilian applications, such as civil aviation. After that, for several reasons, the United States *Department of Defense* (*DoD*) wanted unauthorized users to stop benefiting from high-accuracy GPS service. Therefore, DoD decided that GPS should be developed into a dual-use system which provides two separate types of services, each with its own level of accuracy. The first service was called *Standard Positioning Service* (*SPS*), which was: free of any direct-service fee, dedicated primarily for civil use worldwide, and required relatively low-cost receiving equipment. The second service was called *Precision Positioning Service* (*PPS*), which was intended for military use solely. Therefore, to access PPS, an authorized military-grade receiving equipment was always required ([13] Ch. 1, pp. 3-4).

The DoD thought that via dividing the GPS services into SPS and PPS, they will ensure that the positioning accuracy level attainable to normal users is kept far lower than what is exclusively offered to the authorized personnel. Nonetheless, after extensive testing,

and to the surprise of DoD, the positioning accuracy attained by SPS receivers turned out to be comparable somehow to what should have been offered by PPS receivers exclusively! As a consequence, in what was known as *Selective Availability* (*SA*) in the year 1990, DoD decided to take one extra measure which imposes intentional degradation to the SPS positioning accuracy [14] ([15] Ch. 9, pp. 315-322). In actuality, the undesired SA effect extends further to include PPS as well, but to the contrary of SPS receivers, PPS receivers were equipped to mitigate the undesired SA effect completely. In other words, PPS receivers bypass SA degradation completely as if SA never existed in the first place ([16] Ch. 7, pp. 216). It is worth mentioning, the SA degradation was imposed for almost a decade, and then it was switched off for good in the year 2000. As a matter of fact, there was a significant increase in the accuracy level experienced by normal GPS users once the SA degradation was switched off [17] ([16] Ch. 26, pp. 754). Subsequently, due to the improved accuracy level, mass production of GPS receivers for normal users commenced on an industrial scale, which resulted into a substantial reduction in the cost of GPS receivers, thereby usage of GPS services was widespread among ordinary people on a daily basis [18, 19]. For couple of decades or so, and before the introduction of any viable alternative (or counterpart) to GPS in the civil use market, ordinary people experienced GNSS technology with the help of GPS constellation exclusively. Once the development history of remaining GNSS constellations existing today is presented next, one can get a sense of how other GNSS constellations were far behind GPS in terms of development timeline for civil use.

The first offering of service for civilian applications by the Russian-operated *GLObal'naya NAvigatsionnaya Sputnikovaya Sistema* (*GLONASS*), which is the nearest competitor to GPS in terms of development timeline, came in the year 1988 when it was used for civil aviation. Afterwards, through a decree issued by the Russian government in the year 1995, GLONASS was declared a dual-use system available for civil use worldwide. However, the 1995 decree was, more or less, of a little effect since it turned out that during the late 90s and early 2000s the GLONASS constellation itself was poorly maintained. This resulted into a critical state of only 6 operational satellites whereas a full GLONASS constellation should consist of at least 24 operational satellites [20]. One could argue that thanks to a presidential decree in the year 2007 by Vladimir Putin who is the president of the Russian Federation, the real launch of GLONASS's commercial services commenced on the same decree's year. In fact, Putin backed full restoration of GLONASS constellation and, at the same time, allowed it to become an open service for everyone worldwide [21, 22] ([16] Ch. 8, pp. 219-220).

The remaining two GNSS constellations which have not been covered yet are: the Chinese *BeiDou navigation satellite System* (*BDS*) and the European Galileo. BDS was planned from the beginning to be rolled out in a 3-steps strategy starting from the year 2000. At the time of writing this M.Sc. thesis and according to the website of the Test and Assessment Research Center of China Satellite Navigation Office, there are 33 operational BDS satellites. Once the planned deployment of a full BDS constellation comprising 35 operational satellites is fulfilled around the year 2020, reliable positioning and navigation services for normal users around the world are promised [23]. Finally, the *European Union* (*EU*) endeavour in the realm of GNSS technology ushered in the Galileo constellation, whose space-segment was inaugurated in the year 2005 with the launch of a test satellite named GIOVE-A. At the time of writing this M.Sc. thesis and according to the website of the European GNSS Service Centre, the total number of operational Galileo satellites has reached a new milestone of 22 satellites, that is after commissioning the latest batch of four Galileo satellites for operational use on 11 February 2019. It is expected that Galileo will reach full operational capability during late 2019 ([24] Ch. 1, pp. 3-6).

2.2 Basics of Positioning Using Range Measurements

Unlike the old radionavigation terrestrial systems (e.g., LORAN and Gee) which were based on measuring *time-difference-of-arrival* (*TDOA*), the modern satellite navigation systems are based on measuring *time-of-arrival* (*TOA*) and *time-of-transmission* (*TOT*). During the process of estimating its position, velocity, and/or time, a modern satellite navigation receiver measures both TOA and TOT for a handful of satellite-generated incoming signals. Although labelling satellite navigation systems as being based on TOA (without explicit mentioning of TOT) is slightly misleading, it is, nevertheless, a common practice in navigation systems literature, and hence it is used herein as well. Anyway, in order to estimate the 3D position (longitude, latitude, and altitude) anywhere on the surface of the earth using a system based on measuring TOA, then one way or another, exact distances to three separate points whose locations are precisely known must be acquired. Satellite navigation is not an exception here, and thus the acquisition of exact distances to three separate points (satellites in this case) whose spatial locations are precisely known is a necessary step in the process of estimating the 3D position. To be more specific, navigation using satellites relies basically upon measuring propagation durations of three separate signals from the exact moments they left their respective satellites to the exact moments they were received by the user-segment. Afterwards, by

knowing the fact that the signals were traversing the air interface between the space-segment and the user-segment at the speed of light (denoted as c), then the computed propagation durations could be converted to distances in a straightforward fashion through simple multiplication with c. It is worth mentioning that the involved satellites do provide their precise spatial locations to the user-segment as part of navigation data broadcast. Eventually, after acquiring the exact distances to three separate satellites whose precise spatial locations were provided to the user-segment, the position of the user-segment anywhere on the surface of the earth can be estimated reliably.

Propagation duration is computed practically via subtracting t^{TOT} (which is time-stamped using satellite's clock) from t^{TOA} (which is time-stamped using receiver's clock). However, the time of the receiver's clock is almost always unsynchronized with the time of the corresponding satellite's clock. It is to be noted, the clocks of all satellites belonging to any individual constellation are always perfectly time synchronized with each other. Therefore, due to the momentarily constant time offset T_u, which is present between the clocks of space-segment and user-segment (attributed to asynchronicity as mentioned earlier), the product resulting from multiplying c with the computed propagation duration is not the true geometric distance between the user-segment and the corresponding satellite. Nonetheless, this product resulting from multiplying c with the computed propagation duration is useful and distinctively termed *pseudorange* (which is denoted as ρ) ([6] Ch. 8, pp. 307-309). In a single-constellation positioning, the errors existing in the computed propagation durations between a user-segment and all tracked satellites are always equivalent with each other. As a consequence, by neglecting other error factors for the sake of simplicity (such as the ones presented in *Subsection 2.3.1*), the constant error which is found in every propagation duration computed while tracking any individual constellation's satellite is equivalent to the time offset T_u which is found between the tracked satellites' clocks and the user-segment's clock. Interestingly, the error T_u could be easily detected and eliminated by tracking one more satellite alongside the minimum number of satellites necessary for estimation of a 3D position (which was stated earlier as three satellites). Consequently, in practice, the actual minimum number of satellites which need to be tracked in satellite navigation systems is four satellites. Therefore, a user-segment's 3D position is determined once four distinct pseudorange equations (each one is retrieved from a separate satellite) are solved simultaneously. In the following, a set of four distinct pseudorange equations is shown in *System of Equations 2.1* ([13] Ch. 2, pp. 54).

$$\begin{aligned}
\rho_1 &= c\left(t_1^{\mathrm{TOA}} - t_1^{\mathrm{TOT}}\right) = \sqrt{(x_1 - x_u)^2 + (y_1 - y_u)^2 + (z_1 - z_u)^2} + cT_u \\
\rho_2 &= c\left(t_2^{\mathrm{TOA}} - t_2^{\mathrm{TOT}}\right) = \sqrt{(x_2 - x_u)^2 + (y_2 - y_u)^2 + (z_2 - z_u)^2} + cT_u \\
\rho_3 &= c\left(t_3^{\mathrm{TOA}} - t_3^{\mathrm{TOT}}\right) = \sqrt{(x_3 - x_u)^2 + (y_3 - y_u)^2 + (z_3 - z_u)^2} + cT_u \\
\rho_4 &= c\left(t_4^{\mathrm{TOA}} - t_4^{\mathrm{TOT}}\right) = \sqrt{(x_4 - x_u)^2 + (y_4 - y_u)^2 + (z_4 - z_u)^2} + cT_u
\end{aligned} \qquad 2.1$$

Where:
- c denotes the speed of light in vacuum, which is $299\,792\,458\ [\mathrm{m/s}]$.
- T_u denotes time offset between the clocks of space-segment and user-segment.
- t_n^{TOA} and t_n^{TOT} denote the n^{th} satellite's time-of-arrival and time-of-transmission, respectively.
- x_n, y_n, and z_n together denote the n^{th} satellite's 3D position coordinates.
- x_u, y_u, and z_u together denote the user-segment's 3D position coordinates.

Despite the fact that *System of Equations 2.1* is mathematically nonlinear, a closed-form algebraic solution was derived for the first time in [25]. However, the *System of Equations 2.1* is usually still solved efficiently using iterative numerical techniques based on prior linearization. A satisfactory solution for the *System of Equations 2.1* is the one which reliably deduces the values of the four unknown variables, which are x_u, y_u, z_u, and T_u. To conclude this section, from a receiver architecture point of view, the stage which is tasked with finding a solution for *System of Equations 2.1* is the navigation-processor stage, which is, by the way, extraneous to this M.Sc. thesis.

2.3 Satellite Positioning Error Sources

The positioning accuracy experienced by the end-user is influenced by various error factors, which are often grouped under two classes, namely: *User Equivalent Range Error* (*UERE*) and *Dilution of Precision* (*DOP*). UERE measures the error uncertainty in the computed pseudorange, whilst DOP quantifies the effect left by the satellites geometry on the positioning accuracy ([6] Ch. 9, pp. 424).

2.3.1 User Equivalent Range Error (UERE)

In essence, all GNSS segments (i.e., space-segment, control-segment, and user-segment) contribute one way or another in the final UERE budgets. Also, all error components in the UERE budgets are assumed to be Gaussian-distributed and statistically independent of each other. Therefore, to compute the overall UERE, those error components are usually root-sum-squared together. An example of UERE budgets taken from ([16] Ch. 1, pp. 8-10) is presented in *Table 2.1*. Also, other similar UERE budgets can be

found in ([13] Ch. 7, pp. 321-322), ([6] Ch. 9, pp. 429-431), ([26] Ch. 4, pp. 55-57), and ([27] Ch. 6, pp. 211-215).

Table 2.1: An example of UERE budgets in a standalone GNSS, according to ([16] Ch. 1).

Responsible Segment	**Error Component**	**1σ Contribution** [m]
Space/Control-segment	Broadcasted satellite clock	0.3 − 1.9
	Broadcasted satellite ephemeris	0.2 − 1.0
	Broadcasted group delays	0.0 − 0.2
User-segment	Ionospheric delay	0 − 5.0
	Tropospheric delay	0.2
	Multipath-propagation	0.2 − 1.0
	Receiver noise	0.1 − 1.0
Overall UERE (using root-sum-square)		0.5 − 6.0

By examining *Table 2.1* we can affirm that the biggest contributor in the UERE budgets is the unmodeled ionospheric delay. However, positioning error due to ionospheric delay can be almost totally eliminated (around 98% to be exact) with the help of dual-frequency positioning [28].

2.3.2 Dilution of Precision (DOP)

Due to obstruction of radio signals transmitted by satellites which fall into shallow elevation angle or disappear below the horizon from the standpoint of a user-segment, only a handful of satellites belonging to any individual constellation will find their signals received by the user-segment at the end, and this holds true no matter where the user-segment is geographically located. For instance, the best-case scenario in a GPS constellation comprising 24 satellites in total is 12 in-view satellites, wherein any satellite is considered as in-view from the user-segment perspective in case there was a *line-of-sight* radio link between them. In other words, an in-view satellite is the one which has a physically-unobstructed radio transmission channel with the corresponding user-segment ([16] Ch. 4, pp. 97). It is worth mentioning that power levels are not constant among all signals received from in-view satellites. In fact, received signal's power level varies according to the satellite's elevation angle relative to the user-segment. Specifically, received signal's power level is maximized whenever the satellite is passing overhead (i.e., with steep elevation angle relative to the user-segment). It is recalled herein that any increase in the received signal's power level automatically brings an increase in the *Signal to Noise power-Ratio* (*SNR*). Therefore, the power level of a received radionavigation signal which was transmitted by an in-view satellite passing overhead with respect to the

user-segment is guaranteed to have the largest possible SNR value. However, one should not jump to the conclusion that in terms of satellites geometry, a group of satellites relatively close to one another and at the same time passing overhead with respect to the user-segment is going to bring an excellent positioning accuracy due to the high SNR values anticipated in such scenario. Conversely, this satellites geometry is considered a poor geometry even if it offers high SNR values, it will, in fact, play a detrimental role in the determination of the *Position, Velocity, and Time* (*PVT*) estimates. Even though it is counter-intuitive, but better positioning accuracy is always achieved when the in-view satellites being tracked are well spread apart of each other in the sky with respect to the user-segment [29] ([30] Ch. 8, pp. 130-133).

The question of how much positioning accuracy is lost due to the underlying satellites geometry is typically answered by the DOP. Anyhow, details of how DOP is exactly computed from *System of Equations 2.1* is not presented herein. Nonetheless, what is interesting for us is that DOP is usually divided into several parameters, and the most important parameter is the one termed *Geometric Dilution of Precision* (*GDOP*). In short, GDOP includes error contributions from both of receiver's 3D position and time. As a rule of thumb, the overall positioning error is approximated roughly as the result of multiplying UERE by the GDOP, as shown in *Equation 2.2*.

$$\text{Overall positioning and timing error} \approx \text{UERE} \times \text{GDOP} \qquad 2.2$$

3. CHARACTERISTICS OF GNSS RADIONAVIGATION SIGNALS

3.1 Bit Mapping

GNSS lowpass-equivalent (aka, baseband) signals are usually constructed from a handful of distinct constituent parts, such as: navigation data, PRN spreading codes, and secondary (aka, overlay) codes. All these constituent parts of GNSS baseband signals are inherently digital; in other words, they are not products of *Analog-to-Digital conversion* (*A/D*). Thus, the constituent parts of any GNSS baseband signal are always represented using binary-logic states (i.e., $0/1$, low/high, or on/off), which are referred to as logical-bits from now on. However, it is a common practice in digital telecommunications that sequential blocks of the logical-bit sequence to be mapped into multi-level (and sometimes complex-valued) symbol sequence (denoted as $a[k]$ herein) ([31] Ch. 5). In GNSS, the block size of constituent parts' logical-bits which are consecutively mapped into symbols is equal to one always, thus each symbol conveys one logical-bit only. Also, the symbol alphabet representing the logical-bits is contained within the set $\{-1, +1\}$ ([32] Ch. 2, pp. 10). Using the aforementioned symbol alphabet to represent the logical-bits is known as *Non-Return to Zero* (*NRZ*) polar encoding ([30] Ch. 2, pp. 19). According to what have been discussed so far, there are two possibilities for mapping logical-bit sequences into symbol sequences: either $[0,\ 1] \rightarrow [-1, +1]$ mapping scheme or $[0,\ 1] \rightarrow [+1, -1]$ mapping scheme. Before presenting which mapping scheme is preferable, it is important to understand that GNSS baseband signals are almost always constructed from their constituent parts using bitwise XOR operation (which is equivalent to modulo-2 bitwise addition). Interestingly, the second mapping scheme (i.e., $[0,\ 1] \rightarrow [+1, -1]$) has the following convenient property: bitwise XOR of any two logical-bit sequences can be effectively calculated through symbol-wise multiplication of the same sequences after they have been bit-mapped into symbol sequences. That being said, the second mapping scheme is apparently the preferred one in the GNSS discipline ([8] Ch. 2, pp. 37-41). The mathematical formula which governs the relation between logical-bits and symbols in the second mapping scheme is as shown in *Equation 3.1*.

$$a[k] = 1 - 2\,\tilde{a}[k]; \quad k = 1, 2, \ldots \qquad 3.1$$

Where:

- k denotes a positive integer which is used for sample-indexing throughout this M.Sc. thesis. In order to make k compliant with the notation used in MATLAB (which is a one-based array-indexing language), k always starts from *1.*
- $\tilde{a}[k]$ denotes a logical-bit sequence, where $\tilde{a}[k] \in \{0, 1\}\ \forall k.$
- $a[k]$ denotes a symbol sequence, where $a[k] \in \{-1, +1\}\ \forall k.$

Due to the one-to-one mapping scheme (i.e., one symbol per logical-bit), the symbols generated using the mapping process of *Equation 3.1* are still labeled as bits sometimes in the GNSS literature. Also, due to the one-to-one mapping, the symbol rate is the same as the source logical-bit rate. To conclude this section, if the source logical-bits (the zeros and the ones) of the sequence $\tilde{a}[k]$ were equally likely, or in other words the running disparity is the least possible (i.e., the relation $\left|\sum_k (-1)^{\tilde{a}[k]}\right| \leq 1$ is satisfied), then using the NRZ polar symbol alphabet $\{-1, +1\}$ ensures that the generated symbol sequence $a[k]$ has zero-valued mean ([16] Ch. 4, pp. 99-102).

3.2 Direct-Sequence Spread-Spectrum (DSSS)

The estimation of the synchronization parameters, namely: code-delay, carrier phase, and Doppler-shift in the receiver's signal-processor stage is influenced directly by the characteristics of the radionavigation signals involved. Therefore, some of the signal modulation stages at the satellites have been optimized according to their effect on the receiver's performance while estimating the synchronization parameters. Those modulation stages which have direct effect on the receiver's performance are: the *Direct-Sequence Spread-Spectrum* (*DSSS*) and the *chip pulse-shape* ([16] Ch. 4, pp. 97-107). While DSSS is discussed in this section, chip pulse-shape is left for *Section 3.3*. However, except for pilot (i.e., dataless) signals' channels, GNSS signals have two main functionalities. First functionality, GNSS signals are used for navigation data broadcast. Second functionality, GNSS signals are used for range measurements ([13] Ch. 2, pp. 50-54). As substantiated thoroughly in the continuation of this section, the best technique discovered so far for achieving both aforementioned functionalities is the DSSS technique. In short, one could hardly imagine GNSS technology evolving and reaching its current status without resorting to some form or another of a spread-spectrum technique.

3.2.1 Encoding of Data Messages

The basic principle of a standard DSSS technique is fairly simple: a data message is encoded with a spreading code using a bitwise XOR operation, where the spreading code is generated at a much higher rate than the rate of the data message which is being encoded. It is recalled here that, if mapping process of *Equation 3.1* was implemented, then the symbol-wise multiplication is used instead of the XOR operation, which justifies the use of multiplication in *Equation 3.2*. However, assuming again that the mapping process of *Equation 3.1* was implemented, then the resulting product between the data message and the spreading code belongs to *Binary Phase-Shift Keying* (*BPSK*) constellation alphabet, which is in turn used to modulate an RF carrier signal, such as depicted in *Figure 3.1*. Consequently, due to the capacity of DSSS technique for incorporating data messages in its structure, we can affirm that the first functionality (i.e., GNSS signals are used for navigation data broadcast) is fulfilled when DSSS technique is used to construct the GNSS radionavigation signals.

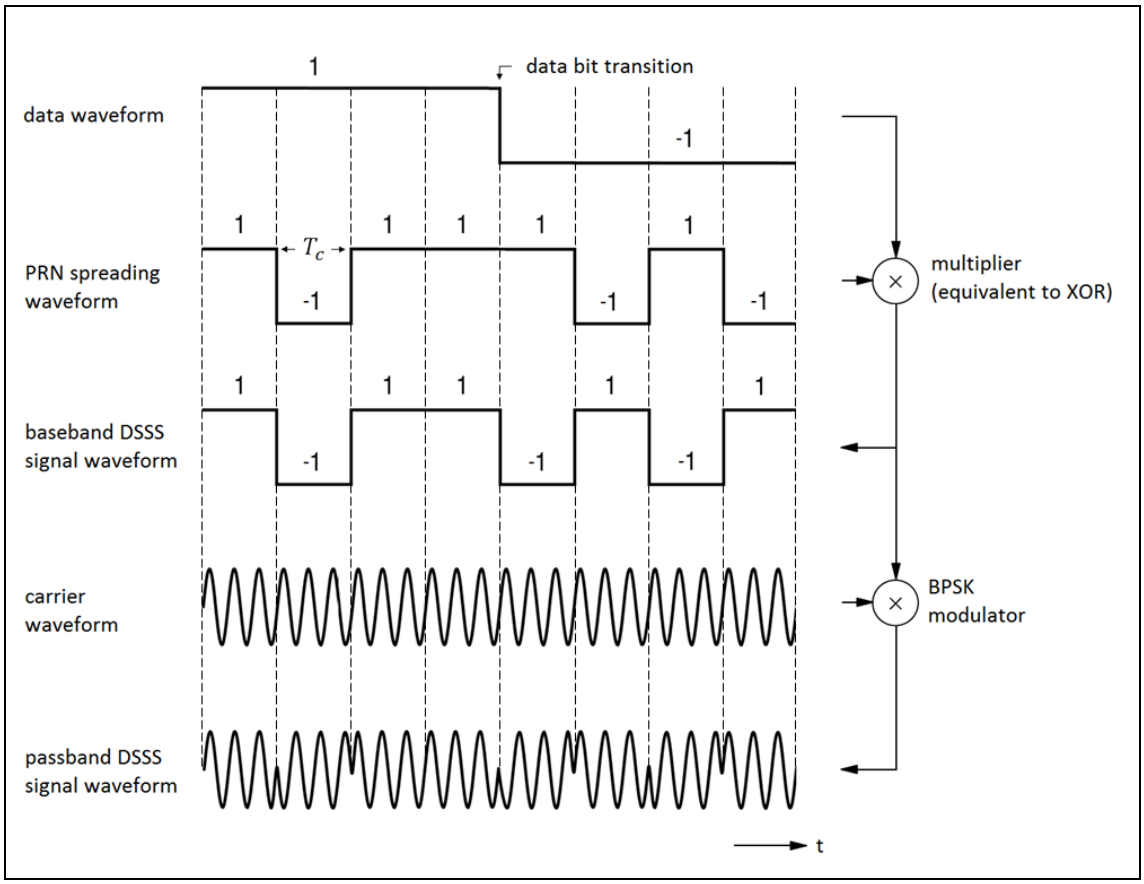

Figure 3.1: Illustration of the encoding process for a data message with a spreading code per DSSS technique. In addition, a carrier modulation process of the resulting baseband DSSS waveform into a passband BPSK waveform is illustrated as well. It is to be noted, rectangular pulse-shape is assumed for both the data message and the spreading code.

3.2.2 Pseudo-Random Noise (PRN) Spreading Code

We turn our attention for the moment to the spreading codes which are commonly used in GNSS signals. Although spreading codes have spectral properties that are similar in principle to random binary noise, they are nevertheless deterministic, hence the term *Pseudo-Random Noise* (*PRN*) spreading codes. To be more specific, those PRN spreading codes are binary *Wide-Sense Cyclo-Stationary* (*WSCS*) with zero-valued mean ([33] Ch. 10). Since the binary-logic states of a PRN spreading code themselves do not represent any information (or data), they are commonly given the distinctive term *chips* instead of the familiar term logical-bits (or bits for short) – by doing so, they are never confused conceptually with the information-bearing logical-bits found in ordinary data messages. The duration of each chip is known as chip-period T_c, whose reciprocal represents the chipping-rate R_c (i.e., $R_c = 1/T_c$). Since R_c is always higher than the bit-rate R_b of the navigation data that is being encoded, the spectrum (or bandwidth) of the resulting DSSS signal is much wider than the spectrum of the navigation data, which, however, justifies the nomenclature “spread-spectrum”. It is worth mentioning, the ratio between the bandwidth of the DSSS signal to the bandwidth of the original navigation data is commonly termed as *processing gain* ([16] Ch. 4, pp. 97-99).

Every PRN spreading code has a predictable periodic pattern, where every repetition-period (denoted as T_{PRN}) is made of the same finite sequence of chips. Thus, the term PRN sequence will be used herein to refer to the chips comprising one of the repetition-periods of a PRN spreading code. While some PRN sequences are reproduced from scratch at the receiver each time they are needed (that is with the help of a certain deterministic formula), other PRN sequences are irreproducible (because they are not defined by any deterministic formula) and thereby they must always be fetched from the memory where they have been stored beforehand ([24] Ch. 3, pp. 39-41). One example of a family of reproducible PRN sequences is the well-known *Gold sequences*, which is one of the earliest family of PRN sequences to be used in the GNSS technology and still being used today. One simple method of producing Gold sequences is through *Linear-Feedback Shift Register* (*LFSR*) circuits [34, 35]. It is worth mentioning that in order to keep the GNSS radionavigation signals running 24/7 nonstop, the PRN sequences must be repeated incessantly without any time gaps. As a concrete example, the PRN sequence in the GPS L1 SPS signal, which is known as *Coarse/Acquisition* (*C/A*), is a Gold sequence of length $1023\ [\mathrm{chip}]$ that is always generated at $R_c = 1.023\ [\mathrm{Mchip/s}]$; hence,

one could easily conclude in this case that the repetition-period T_{PRN} is equal to $1\ [\mathrm{ms}]$. In other words, the PRN spreading code of a GPS L1 SPS signal is a C/A sequence which is generated entirely within one millisecond, then the C/A sequence is repeated all over again in each passing millisecond, indefinitely.

3.2.3 Estimation of Code-Delay

The satellite-generated PRN spreading codes are predefined and configured at the supported receivers; thus, when needed, the receivers can replicate them in an exact manner, leading to what is known as replica signal. Although replicating the PRN spreading codes at the receiver is an important step, it is, nevertheless, not the ultimate goal. In fact, ensuring the replicated PRN spreading code (aka, replica signal) stays aligned (or synchronized) as much as possible with the satellite-generated incoming signal is the sought-after goal in terms of receiver signal-processing. Put simply, the incoming signal and the replica signal are kept synchronized with the help of an operation borrowed from the signal-processing discipline, which is known as *correlation*. Altogether, correlation is the most important operation in the entirety of the receiver's signal-processor stage. If the receiver's signal-processor stage was summarized in a single sentence, then it could be: it is the art of manipulating a replica signal while attempting to align it as much as possible with an incoming signal, primarily with the help of correlation operations, or more precisely, bank of correlators, and subsequently the relevant synchronization parameters are extracted in the process. As depicted in *Figure 3.2*, once the replica signal is aligned as much as possible with the incoming signal, the maximum (aka, peak) of the *Cross-Correlation Function* (*CCF*) is reached, which subsequently uncovers the code-delay Δt that is observed in the satellite-generated DSSS incoming signal by the receiver.

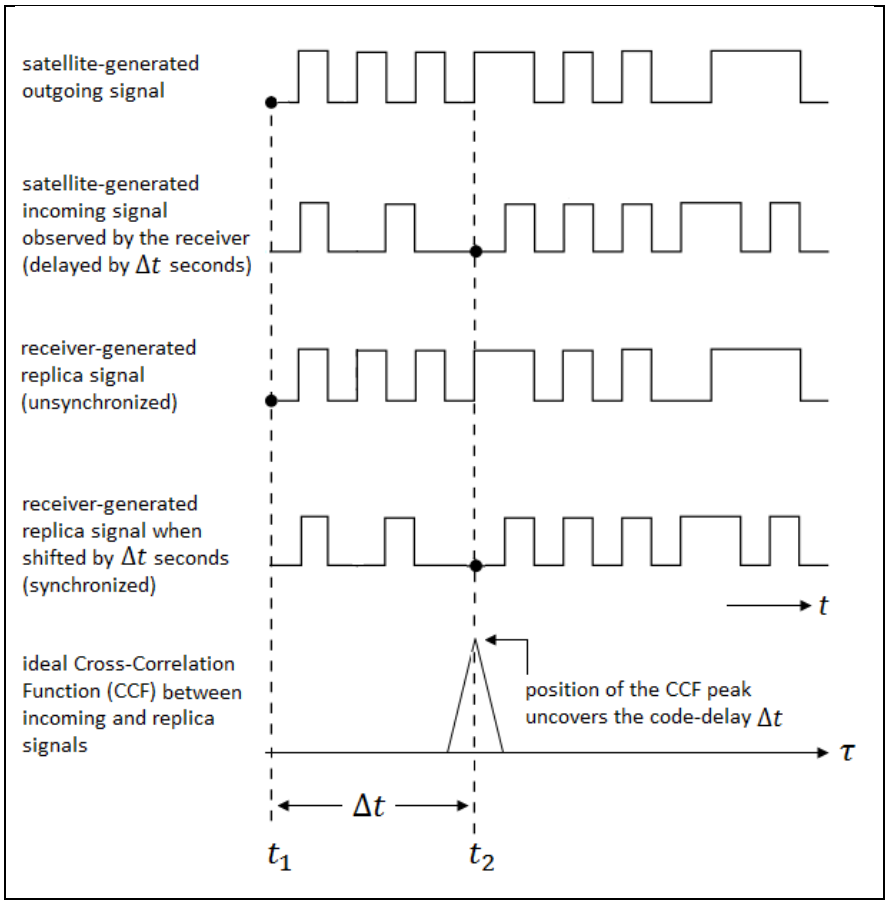

Figure 3.2: Illustration of how the code-delay Δt of a satellite-generated DSSS incoming signal can be measured using the peak of the CCF curve between the incoming signal and its receiver-generated replica.

After showing that correlation operation can uncover the code-delay Δt of a satellite-generated DSSS incoming signal, we can affirm that the second functionality (i.e., GNSS signals are used for range measurement) is fulfilled when DSSS technique is used to construct the GNSS radionavigation signals. Therefore, since the PRN spreading code is an important enabler of range measurement in GNSS, it is sometimes referred to as *ranging-code* in the GNSS literature. However, uncovering the code-delay Δt is an important step forward for computing the range (or pseudorange to be exact) between the receiver and the corresponding satellite. Nonetheless, one should pay close attention that code-delay Δt is not equivalent to the total propagation duration of the radionavigation signal $\left(t^{\mathrm{TOA}} - t^{\mathrm{TOT}} - T_u\right)$, this is assuming the asynchronicity between the satellites' clocks and the receiver's clock is the only source of error. To conclude this section, the role which DSSS technique plays in the GNSS technology is not limited to the two functionalities mentioned so far.

3.3 Chip Pulse-Shape

The symbols comprising the DSSS chip sequence are nothing but a sequence of discrete-time values, which still need conversion to a continuous-time waveform before any further processing can take place. This conversion is usually performed with the help of a special modulation process known as *pulse-shape filtering*. By following the terminology presented in ([31] Ch. 5), we can define the general baseband *Pulse-Amplitude Modulation* (*PAM*) waveform as following: discrete-time data symbols are conveyed through a lowpass transmission channel using continuous-time pulses, where each pulse's amplitude represents the data symbol value. Moreover, the construction of a general baseband PAM waveform is straightforward process, such as: a sequence of discrete-time data symbols is filtered using a valid continuous-time and real-valued pulse-shape. The filtering process is usually carried out using a *convolution operation* (denoted using the asterisk symbol $*$). Anyhow, what concerns us herein is the construction of a baseband PAM waveform which contains the DSSS chip sequence of a GNSS radionavigation signal. Interestingly, this is similar in principle to the general baseband PAM case with the exception that the DSSS chip sequence replaces the discrete-time data symbol sequence, such as given in *Equations 3.2* and *3.3*. Although the term baseband PAM waveform is the accurate term applicable for the DSSS chip sequence which has been converted to a continuous-time baseband waveform using a pulse-shape filtering process, it is, nevertheless, an unusual term within the mainstream GNSS literature ([27] Ch. 3, pp. 59-67) ([16] Ch. 4, pp. 97-99).

$$x[k] = d[k]\, c[k]; \quad k = 1, 2, \ldots \tag{3.2}$$

$$\begin{aligned} x(t) &= g(t) * \sum_{k=1}^{\infty} x[k]\, \delta(t - kT_c) \\ &= \int_{-\infty}^{\infty} g(\tau) \sum_{k=1}^{\infty} x[k]\, \delta(t - kT_c - \tau)\, d\tau = \sum_{k=1}^{\infty} x[k]\, g(t - kT_c) \end{aligned} \tag{3.3}$$

Where:

- $c[k]$ denotes an arbitrary discrete-time PRN spreading code chip sequence, which runs at R_c.
- $d[k]$ denotes an arbitrary discrete-time data symbol sequence, which runs at R_b.
- $x[k]$ denotes a discrete-time DSSS chip sequence, which is generated at R_c.
- $g(t)$ denotes an arbitrary real-valued and continuous-time pulse-shape filter.
- $\delta(t)$ denotes a continuous-time Dirac delta function.
- $x(t)$ denotes continuous-time baseband PAM waveform comprising DSSS chips.

3.3.1 Binary Phase-Shift Keying – Rectangular (BPSK-R)

Regarding the pulse-shape filtering in GNSS signals, as a matter of fact, it is possible to use any viable pulse-shape, and it is also possible to use different pulse-shapes for different DSSS chips. For example, the prominent non-rectangular and time-extended pulse-shape known as *Square-Root Raised-Cosine* (*SRRC*) has been investigated as a feasible pulse-shape for generating GNSS baseband PAM signals ([13] Ch. 4, pp. 115). However, even though spectral efficiency, which is the main advantage of SRRC pulse-shape, is very important from the standpoint of digital transmission, nevertheless strict spectral efficiency is not necessarily desired when it comes to GNSS radionavigation signals ([36] Ch. 3). Therefore, although SRRC pulse-shape was the first candidate among the available pulse-shapes which have been investigated while designing some of the Galileo signals, it was, however, quickly abandoned in favour of the classical and simpler rectangular pulse-shape [37]. In short, implementing an SRRC pulse-shape filtering stage brings extra design complexities; hence, SRRC pulse-shape is not used in any of the existing GNSS radionavigation signals. The rectangular pulse-shape $g_{\sqcap}(t)$, which is used extensively in GNSS radionavigation signals, is presented in time-domain in *Equation 3.4*, also depicted in *Figure 3.3(a)*. Also, the rectangular pulse-shape $g_{\sqcap}(t)$ is time-limited with a period equivalent to T_c, thereby its bandwidth (or spectrum) is extended (infinitely long theoretically) as presented in *Equation 3.6* as well as depicted in *Figure 3.3(c)*. Not to mention, as presented in *Equation 3.5* as well as depicted in *Figure 3.3(b),* the rectangular pulse-shape $g_{\sqcap}(t)$ has a unique *Auto-Correlation Function* (*ACF*) $R_{\sqcap}(\tau)$ that is triangular in shape. As shown previously in *Figure 3.1*, when a rectangular pulse-shape $g_{\sqcap}(t)$ is used to construct a baseband PAM signal, which is in turn used to modulate an RF carrier signal, then the result is a passband PAM signal commonly termed *Binary Phase-Shift Keying with Rectangular pulse-shape* (*BPSK-R*).

$$g_{\sqcap}(t) = \frac{1}{\sqrt{T_c}}\left(U\left(t+\frac{T_c}{2}\right) - U\left(t-\frac{T_c}{2}\right)\right); \quad U(t) = \begin{cases} 1, & \text{when } t \geq 0 \\ 0, & \text{otherwise} \end{cases} \qquad 3.4$$

$$R_{\sqcap}(\tau) = \int_{-\infty}^{\infty} g_{\sqcap}(t)\, g_{\sqcap}^{*}(t-\tau)\, dt = \begin{cases} 1-|\tau|, & \text{when } |\tau| \leq T_c \\ 0, & \text{otherwise} \end{cases} \qquad 3.5$$

$$G_{\sqcap}(f) = |\, \mathcal{F}_t\{\, g_{\sqcap}(t)\,\}\,|^2 = \mathcal{F}_{\tau}\{\, R_{\sqcap}(\tau)\,\} = T_c\, \mathrm{sinc}^2(T_c f) \qquad 3.6$$

Where:

- $U(t)$ denotes Heaviside unit-step function.
- $g_{\sqcap}(t)$ denotes rectangular pulse-shape, whose period is T_c.
- $g_{\sqcap}^{*}(t-\tau)$ denotes a conjugated and time delayed version of $g_{\sqcap}(t)$. Since $g_{\sqcap}(t)$ is real-valued, conjugation is trivial.
- $R_{\sqcap}(\tau)$ denotes Auto-Correlation Function (ACF) of $g_{\sqcap}(t)$.
- $\mathcal{F}_{\tau}\{\cdot\}$ denotes a function which computes Fourier Transform in delay-domain τ.
- $\mathcal{F}_{t}\{\cdot\}$ denotes a function which computes Fourier Transform in time-domain t.
- $G_{\sqcap}(f)$ denotes Power Spectral Density (PSD) of $g_{\sqcap}(t)$.

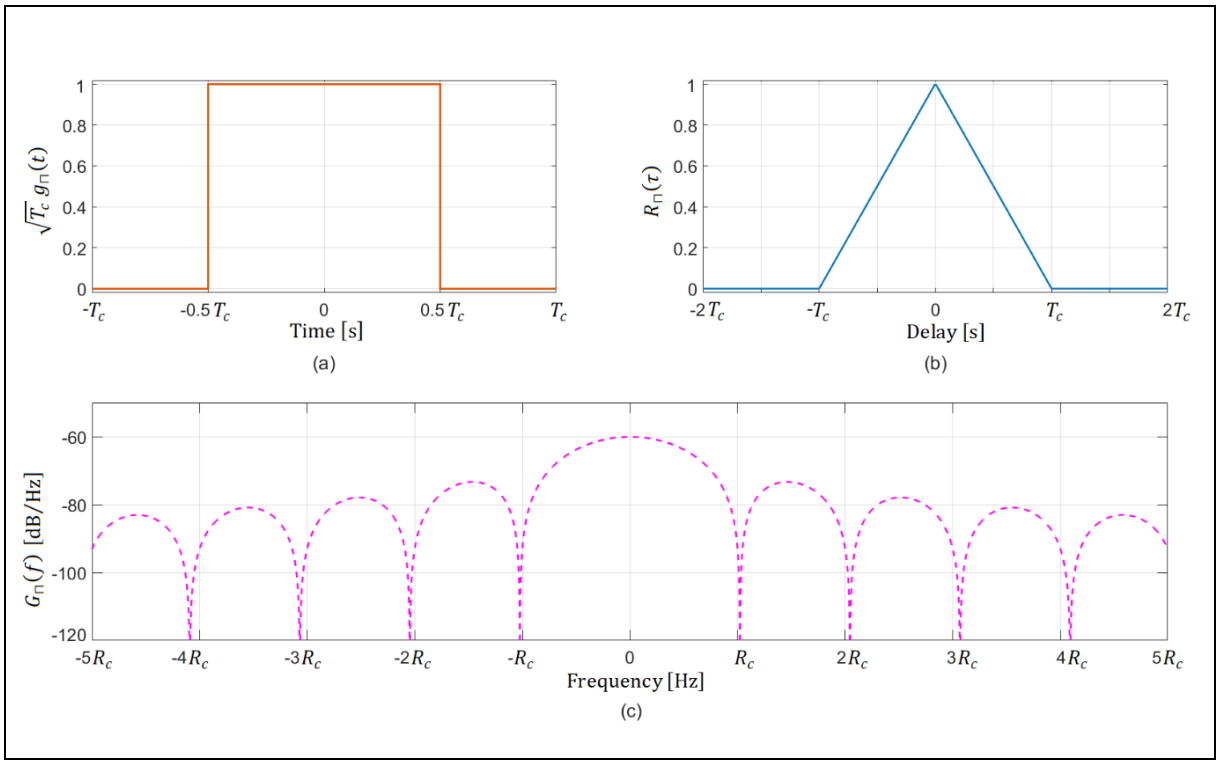

Figure 3.3: (a) ideal rectangular pulse-shape, (b) ACF of the ideal rectangular pulse-shape, and (c) PSD of the ideal rectangular pulse-shape.

3.3.2 Binary Offset Carrier (BOC)

An alternative modulation scheme to BPSK-R is the family of modulations known as *Binary Offset Carrier* (*BOC*). BOC has gained huge acceptance among developers of GNSS signals since its introduction in the year 1999 [38]. Unlike the classical and general purpose BPSK-R which was developed long before the existence of the GNSS technology itself, the relatively new BOC was developed primarily for GNSS radionavigation signals from the beginning. Quoting from the author who is credited with the development of BOC: "modulations designed specifically for radionavigation can outperform existing modulation designs while using the same or even less bandwidth and enabling simple transmitter and receiver designs" [39]. Thus, due to their superior performance, BOC and its derivatives are used extensively in GNSS signals today. BOC scheme is usually defined in the literature using two different models ([36] Ch. 4) ([40] Ch. 2). Herein, we handpick the model which best suits our narrative. Specifically, we handpick the model which treats BOC as a modulation scheme based on a pulse-shape filtering process, and hence *Equation 3.3* is also applicable for generating baseband PAM waveforms of BOC signals. Therefore, the difference between BPSK-R and BOC is in the pulse-shape only. However, theoretically, there are infinite number of possibilities for BOC pulse-shapes; hence, the term $\mathrm{BOC}(n_{sc}, n_c)$ is used to specify the exact form of the pulse-shape in question. As presented in *Equation 3.7,* the pulse-shape $\mathrm{BOC}(n_{sc}, n_c)$ is built basically through breaking up a rectangular pulse (whose period is T_c) into $2n_{sc}/n_c$ smaller rectangular pulses, whose widths are $T_c \div (2n_{sc}/n_c)$ each, and whose amplitudes alternate successively between $+1$ and -1. The break-up process is carried out through multiplication of the rectangular pulse (whose period is T_c) with either a sine or a cosine square wave subcarrier, whose period is denoted as T_{sc} and given as $\boxed{T_{sc} = T_c \div (n_{sc}/n_c)}$. Therefore, the ratio $2n_{sc}/n_c$, which is termed the BOC modulation order [41], represents the number of square wave half-periods per rectangular pulse (the one whose period is T_c), such as depicted in *Figure 3.4(a)* for the case of sine-phased $\mathrm{BOC}(1,1)$. Due to the break-up of rectangular pulses in BOC scheme, the spectrum of the generated waveform, which is shaped by the spectrum of the underlying pulse-shape, is separated into two symmetrical parts, such as given in *Equation 3.8* for the general case, also depicted in *Figure 3.4(c)* for the case of sine-phased $\mathrm{BOC}(1,1)$, thereby BOC modulation is sometimes termed *split-spectrum modulation*. Not to mention, as depicted in *Figure 3.4(b)* for the case of a sine-phased $\mathrm{BOC}(1,1)$, the shape of ACF in BOC scheme is evidently different from that of BPSK-R. Finally, BOC scheme provides spectral isolation to some extent from the legacy GNSS signals while sharing the same carrier frequency with them;

consequently, better *Radio-Frequency Compatibility* (*RFC*) is achieved when BOC scheme is used (for more details refer to *Subsection 4.3.1*).

$$g_{\mathrm{BOC}(n_{sc},\, n_c)}(t) = \begin{cases} g_{n_{sc}}(t)\; g_{n_c}(t), & \text{when } |t| \leq \dfrac{1}{2n_c f_r}\,; \\ 0, & \text{otherwise;} \end{cases} \qquad 3.7$$

$$g_{n_c}(t) = \sqrt{n_c f_r}\left(U\left(t + \frac{1}{2n_c f_r}\right) - U\left(t - \frac{1}{2n_c f_r}\right)\right);$$

$$g_{n_{sc}}(t) = \begin{cases} \mathrm{sgn}\{\, -\sin(2\pi n_{sc} f_r t)\,\}, & \text{when sine-phased} \\ \mathrm{sgn}\{\, -\cos(2\pi n_{sc} f_r t)\,\}, & \text{when cosine-phased} \end{cases}$$

$$G_{\mathrm{BOC}(n_{sc},\, n_c)}(f) = \begin{cases} \dfrac{1}{n_c f_r}\,\mathrm{sinc}^2\left(\dfrac{f}{n_c f_r}\right)\tan^2\left(\dfrac{\pi f}{2n_{sc} f_r}\right), & \text{when } 2n_{sc}/n_c \text{ is even} \\ \dfrac{1}{n_c f_r}\dfrac{\cos^2\left(\dfrac{\pi f}{n_c f_r}\right)}{\left(\dfrac{\pi f}{n_c f_r}\right)^2}\tan^2\left(\dfrac{\pi f}{2n_{sc} f_r}\right), & \text{when } 2n_{sc}/n_c \text{ is odd} \end{cases} \qquad 3.8$$

Where:

- f_r denotes reference frequency, which is always 1.023 [MHz] in GPS and Galileo.
- n_{sc} denotes an integer multiplier of f_r within subcarrier frequency f_{sc} (i.e., $f_{sc} = n_{sc}\, f_r$).
- $\mathrm{sgn}\{\cdot\}$ denotes a signum function, which extracts the sign of its argument.
- $g_{n_{sc}}(t)$ denotes a sine or cosine square wave subcarrier, whose period is given as $T_{sc} = 1/f_{sc} = 1/n_{sc}\, f_r$.
- n_c denotes an integer multiplier of f_r within chipping-rate R_c (i.e., $R_c = n_c\, f_r$).
- $g_{n_c}(t)$ denotes a rectangular pulse-shape, whose period is given as $T_c = 1/n_c f_r$.
- $g_{\mathrm{BOC}(n_{sc},\, n_c)}(t)$ denotes a $\mathrm{BOC}(n_{sc}, n_c)$ pulse-shape.
- $G_{\mathrm{BOC}(n_{sc},\, n_c)}(f)$ denotes the PSD of a sine-phased $g_{\mathrm{BOC}(n_{sc},\, n_c)}(t)$.

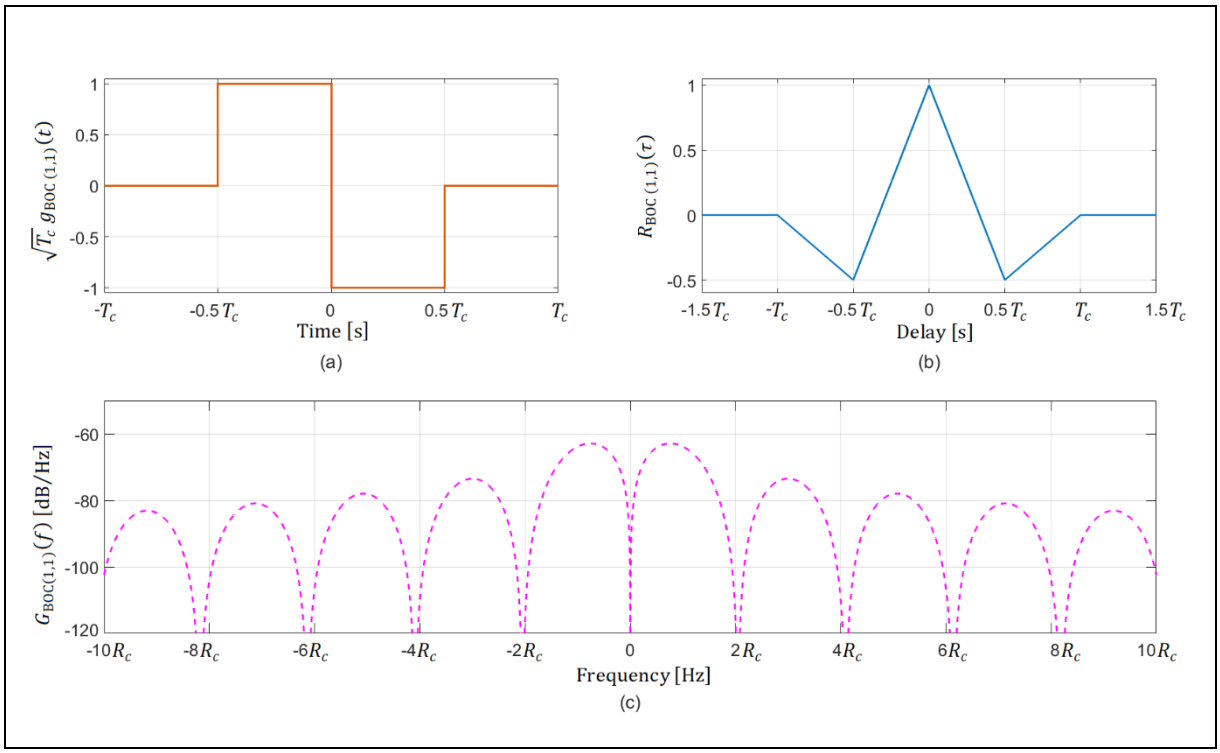

Figure 3.4: (a) ideal sine-phased $\mathrm{BOC}(1,1)$ pulse-shape, (b) ACF of the ideal sine-phased $\mathrm{BOC}(1,1)$ pulse-shape, and (c) PSD of the ideal sine-phased $\mathrm{BOC}(1,1)$ pulse-shape.

3.4 Multiple Access Techniques

There are several well-established multiple-access techniques in the wireless telecommunications discipline, where each one of these multiple-access techniques can allow simultaneous transmission of radio signals from several different terminals across a common transmission channel (i.e., single shared physical medium). Therefore, the wireless transmission channel is usually shared among several simultaneous transmitting terminals; nevertheless, with the help of some specialized multiple access techniques the receiving terminal can still distinguish between the transmitting terminals (and vice versa in case of transceiving terminals). The multiple access techniques commonly implemented in wireless telecommunication systems are classified under four fundamental techniques which are listed below ([42] Ch. 17, pp. 365-366):

- *Frequency-Division Multiple Access* (*FDMA*): transmission channel is divided over multiple disjoint frequency bands.
- *Time-Division Multiple Access* (*TDMA*): transmission channel is divided over multiple separate timeslots.
- *Code-Division Multiple Access* (*CDMA*): each terminal spreads the spectrum of its data via encoding it with a unique and distinctive pseudo-random code.

- *Space-Division Multiple Access* (*SDMA*): radio waves transmitted by different terminals are physically separated, and hence they do not share the same transmission channel.

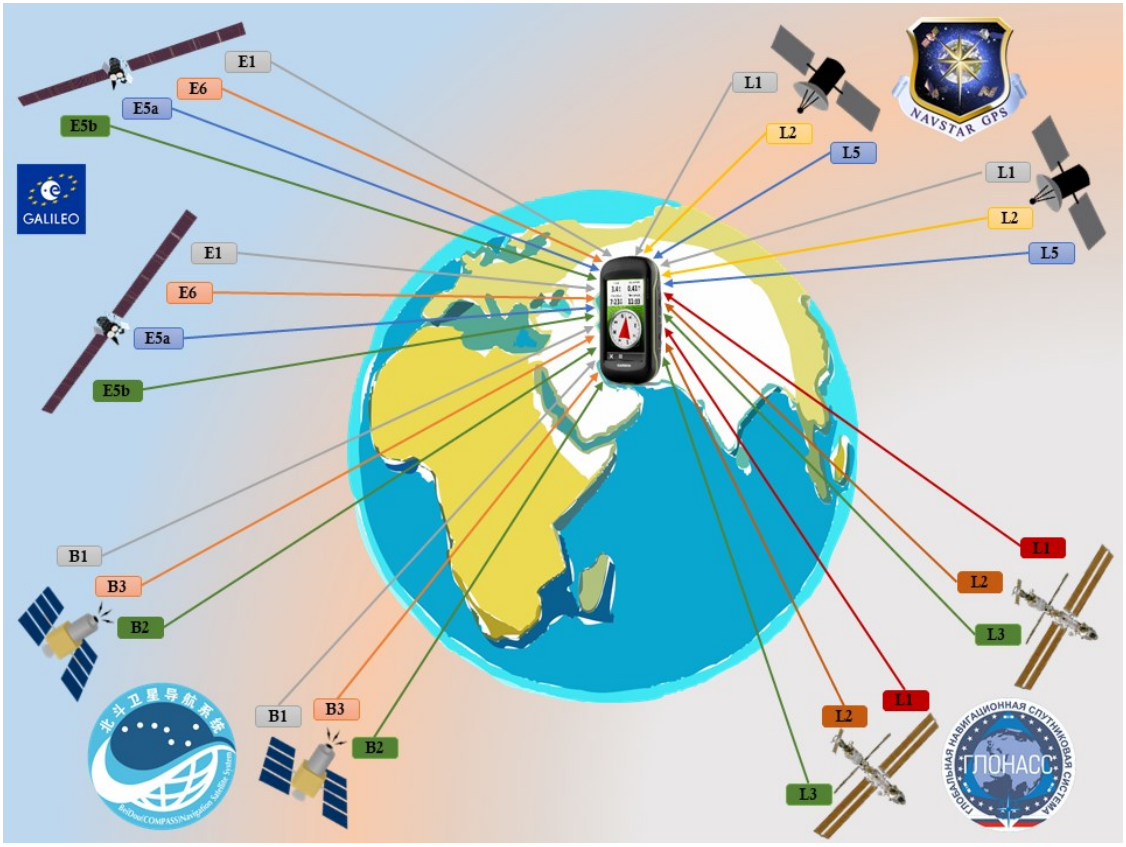

Figure 3.5: An illustration of the air interface between GNSS constellations and user-segment. Signal-plans for all available GNSS constellations are shown in the figure as well. The constellations shown belong to GPS, GLONASS, BDS, and Galileo. Notice, the figure does not show every satellite belonging to each GNSS constellation (only 2 satellites per constellation are shown). Also, signals with similarly coloured blocks/lines are spectrally overlapping.

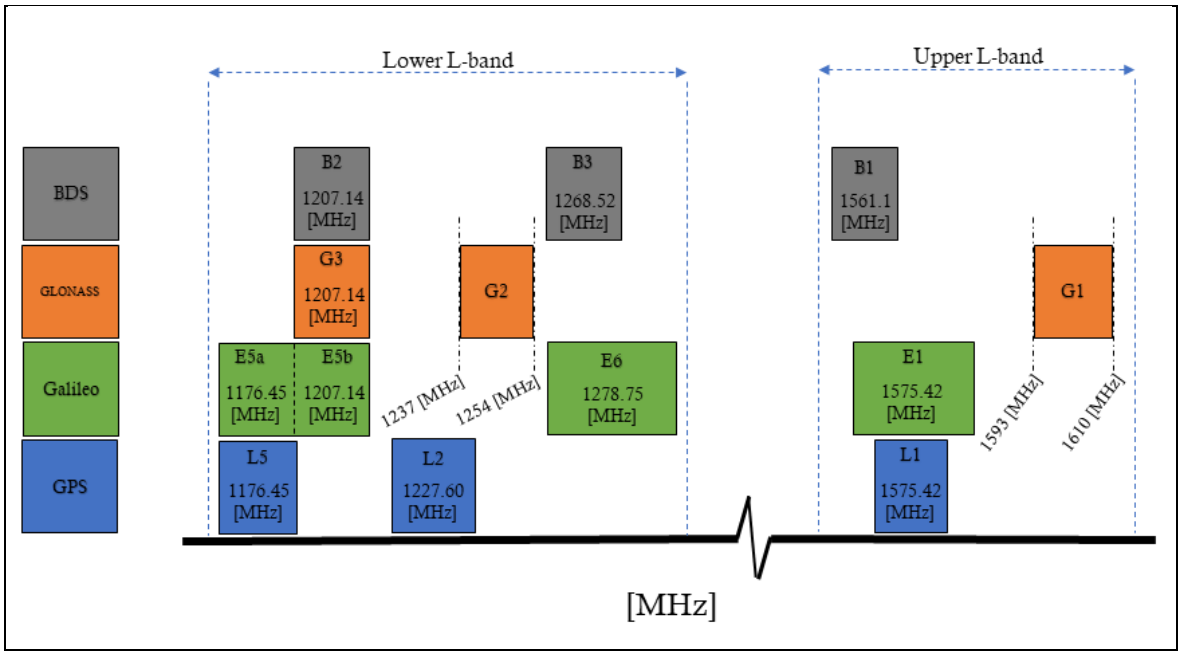

Figure 3.6: An illustration of the signal-plan for GPS, Galileo, GLONASS, and BDS constellations with their respective allocated carrier frequencies. The carrier frequency of each band is given in the figure, except for G1 and G2 bands which contain GLONASS FDMA signals. What is given for GLONASS G1 and G2 is the range of carrier frequencies which are allocated to the sub-bands located within each band, where each sub-band is in turn allotted for a specific satellite (or two antipodal satellites) which belongs to the GLONASS constellation.

It is clear from both *Figure 3.5* and *Figure 3.6* that unless proper multiple access techniques are implemented in GNSS signals, then severe wideband interference will occur caused by both intra-constellation and inter-constellation spectrally overlapping radionavigation signals, which might result into complete inability of receivers to distinguish between different satellite-generated incoming signals. However, as presented in the next two subsections, multiple access techniques are indeed implemented in all GNSS radionavigation signals available today.

3.4.1 FDMA

Every GNSS satellite available today transmits a signal-plan (i.e., a handful of different radionavigation signals), where each signal is carrier-modulated into a designated RF band located within the L-band, which is defined by IEEE as the range of frequencies falling between 1 and 2 [GHz], such as depicted in *Figure 3.6*. However, also from *Figure 3.6*, it can be easily spotted that carrier frequencies are not uniquely allocated among both intra-constellation and inter-constellation signals, so GNSS signals cannot be regarded as FDMA signals. Nonetheless, there are few exceptions here, namely: GLONASS L1SF, L1OF, L2SF, and L2OF legacy signals, which are all contained within the G1 or G2 bands, are considered as pure FDMA signals. To be more specific, each

GLONASS satellite has been assigned a dedicated sub-band, whose bandwidth spans $0.5625\ [\mathrm{MHz}]$, for each of the aforementioned FDMA signals. As a result, the same PRN spreading code is shared across all GLONASS satellite-generated signals which have been transmitted over either G1 or G2 band; nevertheless, the in-view GLONASS satellites are still distinguishable from each other while the receiver is tracking any of their FDMA signals which have been transmitted over G1 band or G2 band. Interestingly, GLONASS operators can save valuable spectrum through exploitation of the situation arising when two satellites are placed on antipodal orbital positions, which allows both satellites to be assigned identical sub-bands within G1 and G2 bands without causing harmful wideband interference between those signals whose carrier frequencies are shared in the process ([16] Ch. 8, pp. 226-229).

3.4.2 DS-CDMA

Moving the discussion to the other multiple access technique known as CDMA, where the energy of each satellite's signal is continuously distributed across the entire available time-frequency plane. As a consequence, the only possible way to let receivers distinguish between two different CDMA signals which have been transmitted simultaneously, onto the same carrier frequency, and using shared transmission channel is to encode each signal with a unique and distinctive pseudo-random code. Luckily, this is achieved automatically while using the DSSS technique given that each DSSS signal of every GNSS satellite was generated using a different PRN spreading code ([13] Ch. 4, pp. 113-115). It is to be noted, the term *Direct-Sequence - Code-Division Multiple Access* (*DS-CDMA*) refers to the case where DSSS technique acts as the basis for CDMA. As a matter of fact, DS-CDMA has many advantages over FDMA when it comes to designing GNSS radionavigation signals. Hence, DS-CDMA is the predominant multiple access technique in the vast majority of GNSS signals available today, even the GLONASS's legacy FDMA signals are being modernized gradually to become purely DS-CDMA signals. Most important disadvantage for FDMA against CS-CDMA, is the increased design complexity of receiver frontends which are dedicated for FDMA signals. Finally, as far as today's GNSS signals are concerned, TDMA and SDMA have no role to play whatsoever. Nonetheless, SDMA was proposed in [43] as a feasible technique for fixing the *Multiple-Access Interference* (*MAI*) issue which is likely to emerge in prospective OFDM-based satellite-generated radionavigation signals.

3.5 Summary of Characteristics for Relevant GPS and Galileo Signals

GNSS signals are genuinely different from one another in many respects, and this holds true even in case of intra-constellation signals. In this section, we give a tabulated general overview of the most important characteristics defining the radionavigation signals which are of relevance for us in this M.Sc. thesis. It is to be recalled that, herein we care the most about four different GNSS radionavigation signals, namely: Galileo E1, and Galileo E5, GPS L1, and GPS L5. The information presented in *Table 3.1*, *Table 3.2*, *Table 3.3*, and *Table 3.4*, is gathered from [6, 13, 16, 32, 34, 36, 44].

Table 3.1: Characteristics of Galileo E1 radionavigation signal.

Signal	Galileo E1		
Channel	E1-A	E1-B (data)	E1-C (pilot)
Service *	PRS	OS, SoL, and CS	
Carrier Frequency †	$154 \times f_0 = 1575.42$ [MHz]		
Minimum Received Power Level Above 10º Elevation Angle	Classified	-160 [dBW] per channel per satellite -157 [dBW] per B/C composite signal per satellite	
Modulation Scheme	Cosine-phased BOC(15,2.5)	CBOC(+)	CBOC(−)
		CBOC(6,1,1/11) for the B/C composite signal	
Carrier Component	quadrature	in-phase	
Subcarrier Frequency	15.345 [MHz]	1.023 [MHz] and 6.138 [MHz]	
Multiple Access	DS-CDMA		
PRN Spreading code	Classified	Memory code, consists of 4 092 [chip] per channel per satellite	
		Chipping-rate is 1.023 [Mchip/s]	
	Chipping-rate is 2.56 [Mchip/s]	Repetition-period is 4 [ms]	
Secondary (aka, Overlay) Code	Classified	N/A	Consists of 25 [chip]
			Runs at 250 [chip/s]
Navigation Data ‡	Classified	I/NAV structure	N/A
		Bit-rate is 250 [bit/s] after 1/2-rate FEC	

Table 3.2: Characteristics of Galileo E5 radionavigation signal.

Signal	Galileo E5			
Sideband	E5a lower-sideband		E5b upper-sideband	
Channel	E5a-I (data)	E5a-Q (pilot)	E5b-I (data)	E5b-Q (pilot)
Service *	OS and CS		OS, SoL, and CS	
Carrier Frequency†	$115 \times f_0 = 1176.45$ [MHz]		$118 \times f_0 = 1207.14$ [MHz]	
	1191.795 [MHz] for the full-band signal			
Minimum Received Power Level Above 10º Elevation Angle	−158 [dBW] per channel per satellite −155 [dBW] per sideband per satellite −152 [dBW] per full-band per satellite			
Modulation Scheme**	BPSK-R(10) per channel QPSK with rectangular pulse-shape per sideband AltBOC(15,10) per full-band signal			
Carrier Component	in-phase	quadrature	in-phase	quadrature
Subcarrier Frequency	15.345 [MHz]			
Multiple Access	DS-CDMA			
PRN Spreading code	M-seq, consists of 10 230 [chip] per channel per satellite			
	Chipping-rate is 10.23 [Mchip/s]			
	Repetition-period is 1 [ms]			
Secondary (aka, Overlay) Code	Consists of 20 [chip]	Consists of 100 [chip] per satellite	Consists of 4 [chip]	Consists of 100 [chip] per satellite
	Run at 1 [Kchip/s]			
Navigation Data‡	F/NAV structure Bit-rate is 50 [bit/s] after 1/2-rate FEC	N/A	I/NAV structure Bit-rate is 250 [bit/s] after 1/2-rate FEC	N/A

Table 3.3: Characteristics of GPS L1 radionavigation signal.

Signal	GPS L1	
Channel	Coarse Acquisition (C/A)	P(Y)
Service*	SPS	PPS
Carrier Frequency†	$154 \times f_0 = 1575.42$ [MHz]	
Minimum Received Power Level††	−158 [dBW] per satellite	−161.5 [dBW] per satellite
	Power of the composite signal is −156.7 [dBW] per satellite	
Modulation Scheme**	BPSK-R(1)	BPSK-R(10)
Carrier Component‡‡	quadrature	in-phase
Multiple Access	DS-CDMA	
PRN Spreading code	Gold code, consists of 1023 [chip] per satellite	Encrypted M-seq, consists of 6 187 104 000 000 [chip] per satellite
	Chipping-rate is 1.023 [Mchip/s]	Chipping-rate is 10.23 [Mchip/s]
	Repetition-period is 1 [ms]	Repetition-period is 7 [day]
Navigation Data	LNAV structure	
	Bit-rate is 50 [bit/s]	

Table 3.4: Characteristics of GPS L5 radionavigation signal.

Signal	GPS L5	
Channel	L5-I (data)	L5-Q (pilot)
Carrier Frequency†	$115 \times f_0 = 1176.45$ [MHz]	
Minimum Received Power Level††	-157 [dBW] per channel per block IIIA satellite -154 [dBW] per I/Q composite signal per block IIIA satellite	
Modulation Scheme	QPSK with rectangular pulse-shape for the combined signal	
Carrier Component	in-phase	quadrature
Multiple Access	DS-CDMA	
PRN Spreading code	M-seq, consists of $10\,230$ [chip] per channel per satellite	
	Chipping-rate is 10.23 [Mchip/s]	
	Repetition-period is 1 [ms]	
Secondary (aka, Overlay) Code	Neumann-Hoffman (N-H) codes	
	Consists of 10 [chip]	Consists of 20 [chip]
	Run at 1 [Kchip/s]	
Navigation Data‡	CNAV structure	N/A
	Bit-rate is 100 [bit/s] after 1/2-rate FEC	

* *Standard Positioning Service (SPS), Precision Positioning Service (PPS), Public Regulated Service (PRS), Open Service (OS), Safety of Life (SoL), and Commercial Service (CS).*

† f_0 *denotes the fundamental frequency, which is always* 10.23 [MHz] *in GPS and Galileo at least.*

‡ *FEC stands for Forward Error Correction.*

** *BPSK-R*(n) *denotes a signal which is modulated using BPSK-R scheme, whose chipping-rate* $R_c = n \times f_r$*, where* f_r *in turn denotes the reference frequency which is always* 1.023 [MHz] *in GPS and Galileo at least.*

†† *These power levels are applicable for: "satellites at or above* 5° *elevation angle, while assuming the user-segment is located on or near the surface of the Earth with a* 3 [dBi] *linearly polarized antenna at worst normal orientation" ([16] Ch. 7, pp. 209).*

‡‡ *"Terminology of 'in-phase' and 'quadrature' is used only to identify the relative phase quadrature relationship of the carrier components (i.e.,* 90° *offset of each other)" ([34] Ch. 3, pp. 15).*

4. INTEGRATION ISSUES AMONG EXISTING GNSS CONSTELLATIONS

4.1 Introduction to Compatibility and Interoperability

It is worthwhile to mention in the beginning of this chapter, the term “GNSS operator”, which is used repeatedly throughout this chapter, does not refer to a conventional operator in the strict sense of the word. For example, the concept of a GNSS operator is different in many respects from the concept of a typical mobile telephone network operator. In addition, the operators of the existing GNSS constellations are different from one another in terms of organizational structure. Hence, it is difficult to give a general definition of what exactly constitutes a GNSS operator. Nonetheless, there is a common factor among all existing GNSS operators, which is the ownership by the government of the country of origin. For instance, once we track the hierarchical organization of the GPS operator, we find the federal government of the United States as the highest level of authority, such as following: GPS is operated by the *Air Force Space Command* (*AFSPC*), which is in turn a major command of the *United States Air Force* (*USAF*), which is in turn a military service branch organized within the *Department of the Air Force* (*DAF*), which is in turn one of the three subordinate military departments within the *Department of Defense* (*DoD*), which is in turn an executive branch department of the federal government. On the other hand, GLONASS is operated by *Roscosmos State Corporation for Space Activities*, which is the national space agency of the Russian Federation. Similarly, BDS is operated by *China’s National Space Administration* (*CNSA*), which is the country’s national space agency. However, there is a subtle difference here in the case of Galileo constellation, wherein the ownership of the operator does not belong to a single government. Instead, the ownership of Galileo constellation belongs to a group of governments which runs the coalition of countries known as EU. To be more specific, at the time of writing this M.Sc. thesis, the operator which is responsible for Galileo constellation is the intergovernmental organization known as *European GNSS Agency* (*GSA*). It should be noted, the topic of who finance, design, develop, build, govern, operate, administer, maintain, upgrade, advertise, market, etc. each of those existing GNSS constellations is a much more involved topic than what has been presented already in this paragraph. Therefore, even though it is just a scratch on the surface, we settle for what has been presented so far for the sake of conciseness.

As we have moved into an era where GNSS technology plays a significant role in shaping many aspects of modern-day civilization, several developed nations have decided to pursue their ambitions of creating their own versions of GNSS technology. Consequently, GNSS technology has been evolving for the past couple of decades all the way to the current status where multiple independent constellations are working simultaneously on parallel with each other. But in order to ensure seamless coexistence between such diversity of GNSS constellations, continued bilateral and multilateral cooperation must be kept always between the different parties involved. Most importantly, harmful interference induced by one constellation to the functionalities of other existing constellations must be kept in check. It is recalled here that all GNSS satellites share the same wireless transmission channel known as air interface. In addition to that, letting some GNSS signals which are transmitted by satellites belonging to the same constellation alongside satellites belonging to other existing constellations to share the same carrier frequency (or to overlap partially in terms of spectrum) across the air interface (as seen in both *Figure 3.5* and *Figure 3.6*), is commonly practiced in GNSS technology. Therefore, intra-constellation and inter-constellation *Radio Frequency Interference* (*RFI*) is prevalent in GNSS signals, and hence it must be investigated carefully in order to keep it from reaching intolerable levels. In fact, compatibility, or more precisely *Radio-Frequency Compatibility* (*RFC*), whose main purpose is to keep RFI in check, is a key issue in GNSS technology and under no circumstances should it be violated by any GNSS operator.

GNSS operators are not expected to restrict their mutual cooperation merely at achieving compatibility, instead they are determined to go one step further and deliver interoperability between their respective constellations ([8] Ch. 3, pp. 86-88). It is worth mentioning, the existence of interoperability among different GNSS constellations is the key for achieving multi-constellation positioning, which is an increasingly desired feature in nowadays GNSS receivers. To be more specific, designing a receiving equipment capable of multi-constellation positioning without the existence of interoperability is going to be an intricate task to say the least. Although the operators of GNSS constellations have decided to develop their technologies independently and in their own way in some respects, they have, nevertheless, sought collaboration with one another in many other respects. In other words, unlike some operators of other cutting-edge technologies who have repeatedly shown tendency to seek competition, dominance, and/or confrontation with their technologies, all current GNSS operators have fortunately opted to collaborate with one another for the sake of maintaining a satisfactory level of compatibility and interoperability between their respective constellations. Collaboration among such variety

of GNSS operators should ultimately deliver an enhanced positioning service to the end-user while keeping the service freely accessible to everyone without any direct-service fee for the most part. However, interoperability demands that signals from different constellations share the same carrier frequency, which is unfavourable in terms of RFC; hence, a trade-off situation between interoperability and RFC arises here. Specifically, interoperability comes with the induced penalty of raising RFI level among GNSS signals. As discussed in the continuation of this chapter, in order to maintain an acceptable level of RFC while concurrently preserving interoperability with all its added benefits, a compromise must be reached first. Eventually, in order to ultimately reach the goal of interchangeability, which is the utmost level of integration possible among GNSS constellations, as seen in *Figure 4.1*, it is believed that issues arising from compatibility and interoperability among different GNSS constellations deserve more attention in future studies.

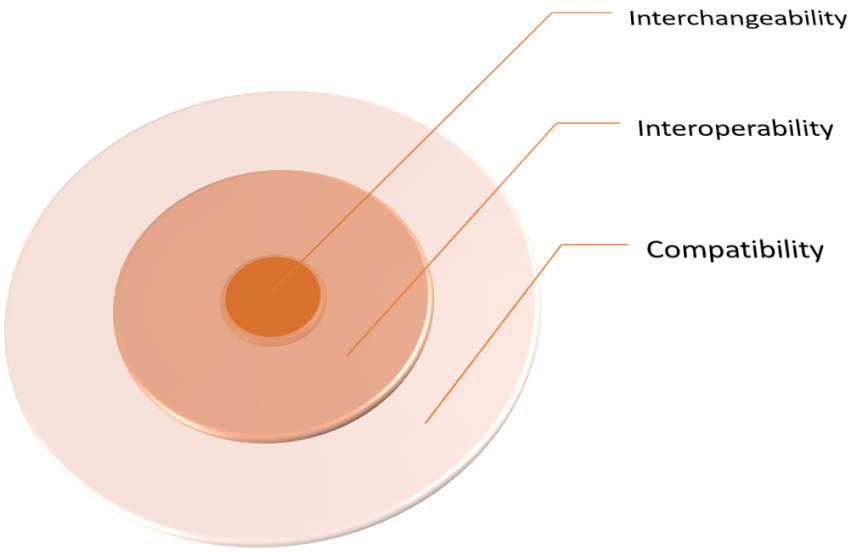

Figure 4.1: A chart illustrating the hierarchical levels of possible integration among different GNSS constellations.

4.2 Radio-Frequency Compatibility (RFC)

As a starting point for this section, the following question is asked: are different GNSS constellations able to coexist seamlessly together without harmful RFI between their transmitted radionavigation signals? This question will be answered gradually in the continuation of this section. Anyhow, as was mentioned earlier in *Section 4.1*, RFC in GNSS

is hindered by two types of wideband RFI, namely: intra-constellation and inter-constellation. To be more specific, intra-constellation RFI is the type of air interface interference that exists among radionavigation signals whose carrier frequencies are shared (or whose spectra are partially overlapping at least) which are transmitted by satellites belonging to the same constellation, and it is commonly termed *Multiple-Access Interference - intrasystem* (*MAI-A*). On the other hand, inter-constellation RFI is similar to MAI-A, but in this case, it exists among radionavigation signals which are transmitted by satellites belonging to separate constellations, and it is similarly termed *Multiple-Access Interference - intersystem* (*MAI-R*). The term MAI-A/R will be used herein to refer to either MAI-A, MAI-R, or both types of interference. Anyway, one should not be too pessimistic about MAI-A/R in GNSS signals, especially after knowing that wideband RFI between the various GNSS signals was investigated and handled carefully when the signals were designed in the first place. Specifically, DS-CDMA scheme whose main purpose is to let various radio signals coexist together on the same transmission channel while preventing them from corrupting each other, is widely implemented throughout GNSS signals. Therefore, the GNSS signals which are based on DS-CDMA scheme are inherently equipped to withstand MAI-A/R to a large extent. It is to be recalled, the underlying structure for the GLONASS radionavigation signals existing in G1 or G2 band is of DSSS type, which itself is the basis for the DS-CDMA scheme as well (for more details refer to *Subsection 3.4.2*). Nonetheless, in the case of GLONASS radionavigation signals existing in G1 or G2 band, the same PRN spreading code is shared among all DSSS signals contained within each band. Thus, for the sake of enforcing intra-constellation multiple access among all GLONASS's radionavigation signals existing in either G1 or G2 band, DSSS technique is carried out in conjunction with a conventional FDMA scheme. As a matter of fact, GLONASS radionavigation signals existing in G1 or G2 band are, in particular, the only GNSS radionavigation signals available today which are not regarded as DS-CDMA signals (for more details refer to *Subsection 3.4.1*). However, even though GLONASS's G1 and G2 bands contain FDMA signals exclusively, these signals are not less immune to MAI-A/R than the widespread DS-CDMA signals by any means. In fact, due to the variation in the carrier frequencies, signals contained within GLONASS's G1 and G2 bands are better protected against intentional man-made narrowband RFI ([16] Ch. 8, pp. 226-229).

From the previous paragraph, it is concluded that signals transmitted by GNSS constellations, regardless if they are DS-CDMA or FDMA signals, are intrinsically RF compatible with each other to a large extent. Nonetheless, DS-CDMA which is the dominant multiple

access scheme in GNSS signals is not the silver bullet against all MAI-A/R problems. In fact, DS-CDMA scheme still has its own limitations in terms of preventing MAI-A/R completely. Consequently, wideband RFI in GNSS signals is still detrimental factor even with the DS-CDMA scheme in place, so we must exercise some caution here while treating the RFC topic in GNSS. It should be noted, the performance of GNSS receivers deteriorates dramatically due to harmful MAI-A/R, especially in the case of large mismatch at received power levels of spectrally overlapping DS-CDMA signals which have been transmitted by different satellites. Moreover, this phenomenon of performance deterioration due to variation in received power levels is commonly referred to as the *near-far problem*. A rigorous discussion about some techniques available to deal with the infamous near-far problem in GPS receivers is found in [45]. It is worthwhile to mention, there is a special type of wideband interference which is confined to super-heterodyne receivers, especially the ones designed for multi-frequency positioning (by the way, it is dual-frequency in most cases). This special type of wideband interference is effectively similar to the MAI-A/R, except that it is not created at the air interface as in the MAI-A/R case. To be more specific, the analog frontend of super-heterodyne receivers which are capable of multi-frequency positioning is designed to downconvert all signals encountered in the designated disjoint bands from RF down to a shared *Intermediate Frequency* (*IF*). Consequently, a new type of interference arises here, specifically when signals which have been transmitted originally over disjoint bands are mixed together using the same IF carrier inside the receiver. Therefore, even though such signals were initially prevented from interfering with each other when they were traversing the air interface, they will eventually interfere with each other inside some types of receivers ([24] Ch. 4, pp. 78-82). Most importantly, RFC is concerned with the interference that is exerted at air interface only, and hence any interference which is produced exclusively at the receiver due to common IF mixing is excluded from RFC considerations.

With the increasing number of GNSS satellites in the sky, the radio spectrum of the L-band has become more congested with signals, which should automatically lead to a degraded RFC ultimately. Therefore, performance deterioration due to wideband RFI has been investigated repeatedly in the GNSS literature. For instance, the *Spectral Separation Coefficient* (*SSC*) was defined in [46] as a figure of merit which quantifies the severity of RFI inflicted upon a GNSS radionavigation signal when other radionavigation signals share the same band with it. To be more specific, in [47], it is stated that SSC criterion is appropriate for measuring the impact left by the interfering signals on the

receiver processing stages which are confined to use exclusively the Prompt-replica correlator (aka, punctual correlator), namely: signal acquisition, carrier tracking loop, and navigation data demodulation. Also in [47], a modified RFC criterion named *Code Tracking - Spectral Sensitivity Coefficient* (*CT_SSC*) is defined and tested under various wideband RFI scenarios. It is worth mentioning, CT_SSC is able to quantify the detrimental role wideband RFI plays in the receiver's code tracking loop, which does not generally employ the Prompt-replica correlator as in the case of carrier tracking loop. To be more specific, the correlators commonly used in the code tracking loop are the Early-replica and the Late-replica. Interestingly, it was shown in [47] that in case of severe MAI-A/R, CT_SSC indicates up to $7\ [\mathrm{dB\text{-}Hz}]$ of degradation in the effective *carrier-to-noise-density-power-ratio* (C/N_0). Additional simulation results concerning SSC and CT_SSC coefficients are presented in [48], where they have been analyzed through various scenarios of wideband RFI. To conclude this section, even some of the worst-case RFI degradation scenarios are still tolerable according to the standards of most GNSS receivers [49].

4.3 Interoperability and Multi-Constellation Positioning

In this section, while emphasizing its relationship to multi-constellations positioning, we move the discussion to the interoperability issue which arises among different GNSS constellations. In GNSS literature, interoperability is usually divided into two levels, namely: system-level and signal-level. Put simply, any two separate GNSS constellations are regarded as system-level interoperable if both constellations can separately provide the same PVT estimates within the accuracy limits set by each individual constellation. This is applicable, for instance, in receivers capable of multi-constellation positioning which are working in single-constellation mode [50]. Thus, if system-level interoperability exists, then any constellation can autonomously provide the PVT estimates without taking any external aid from other existing constellations. From the previous argument, we conclude that system-level interoperability is chiefly concerned with redundancy, which in general ensures an added robustness to the positioning service being offered to end-users. On the other hand, defining signal-level interoperability is a trickier task; nevertheless, it is still presented next. Before defining signal-level interoperability, we affirm that system-level interoperability exists always among all current GNSS constellations, namely: GPS, GLONASS, BDS, and Galileo. Therefore, it is up to the user-segment to choose from which constellation to obtain PVT estimates. Anyway, back to signal-level interoperability, it is primarily a responsibility of the GNSS operators if they want their constellations to be signal-level interoperable with each other or not. In fact,

any two GNSS constellations are regarded as signal-level interoperable when both constellations conform strictly to the rules of interoperability, which some of them are thoroughly covered later in this section. From this point onwards, unless system-level interoperability is mentioned explicitly, then interoperability (when mentioned alone) always refers to signal-level interoperability. However, our next task is to delve into the details of interoperability in order to understand these rules of interoperability which are still ambiguous until now. Before discussing rules of interoperability in details, we shed some light on multi-constellation positioning while focusing on its relationship to interoperability in the next paragraph.

Starting with a quick glimpse into some of the processing stages involved in receivers capable of multi-constellation positioning. First, simultaneous tracking of signals coming from multiple constellations which are of interest to the user-segment is carried out. Then, the measurements retrieved from the multiple constellations being tracked by the user-segment are combined into a single system of observation equations, which are either pseudorange or carrier-phase equations. Eventually, the system of observation equations is solved, which results into determination of the hybrid PVT estimates [50]. It is worthwhile to mention, the first known attempt by any private company to design a receiving equipment capable of multi-constellation positioning which combines measurements from both GPS and GLONASS constellations was a prototype developed by Magnavox company in the year 1990. Furthermore, duo to the total lack of interoperability between GPS and GLONASS constellations back then, the prototype developers had to compensate for all of the following: signals characteristics, carrier frequencies, data bit-rates, navigation data structures, filters delays, and coordinate/time reference frames, all of which within the receiver prototype [1]. Now the question is, what is so special about multi-constellation positioning (dual-constellation in this case) that made it worth the effort of the engineers who were involved in the intricate task of designing that Magnavox receiver prototype? The answer of this question could be found for instance in [20, 21], which is summarized in the following (till the end of this paragraph). Taking advantage of any two or more independent GNSS constellations through simultaneous combining of their respective observation measurements in the receiver, leads to a noticeable increase in the number of in-view satellites in the sky at any point on the surface of the earth and at any moment during the day. This is, however, an obvious result deduced from the simple fact that there are more in-view satellites whenever there are more constellations accessible to the same user-segment. It should be noted, from user-segment's perspective, any increase in the number of in-view satellites should automatically

lead to a satellites geometry which is more spread out in the sky. It is to be recalled here, positioning accuracy increases spontaneously whenever the geometry of the satellites in question is more spread out in the sky from the standpoint of user-segment (for more details refer to *Subsection 2.3.2*). In addition, there is a higher availability of the minimum number of in-view satellites necessary for determining the PVT estimates (which is known to be minimum 4 in-view satellites) when multi-constellation positioning is used. Moreover, this higher availability is achieved even at unfavourable environments where only a portion of the sky is visible, for example in urban areas, mountainous regions, or canyons. Eventually, the PVT estimates integrity is increased, also the determination of the integer ambiguity in carrier-phase observable measurements is accelerated.

From the previous paragraph, we can conclude that multi-constellation positioning has quite a few noteworthy benefits. But is it true that multi-constellation positioning comes always with a penalty, specifically an increase in the design complexity of the receiving equipment as was mentioned earlier in the case of Magnavox receiver prototype? This question is answered as following: unless interoperability exists among the different GNSS constellations which are of interest to the user-segment, then yes, implementing multi-constellation positioning will always bring an added design complexity to the receiving equipment. Therefore, we conclude that interoperability among GNSS constellations strives to deliver a robust positioning solution while eliminating at the same time the need for extra design complexities in the receiving equipment [51]. However, in order to ensure interoperability among different GNSS constellations, some specific bilateral agreements must first be reached between the various parties involved in operating the constellations. Moreover, multilateral agreements through the *International Telecommunication Union* (*ITU*) and the *International Committee on GNSS* (*ICG*) must be reached as well [1]. Back to the rules of interoperability, in order to ensure interoperability among different GNSS constellations, there are certain attributes of each constellation which must be addressed according to what is presented in the following three subsections ([24] Ch. 2, pp. 30-32).

4.3.1 Signals-in-Space

It is important to distinguish that some of the signal's characteristics do not affect interoperability whatsoever. For example, as far as interoperability is concerned, it does not matter if signals from different constellations conform to one another in terms of modulation, waveform, PRN spreading code, and/or navigation data format. Thus, GNSS operators are free to decide on the aforementioned characteristics for their signals without

any conceivable damage to interoperability. On the other hand, if the GNSS constellations are to be interoperable, then the operators are obliged to let some of their signals share the same RF band (in fact, the same carrier frequency). It should be noted, sharing the same carrier frequency among different GNSS signals is needed in order to facilitate carrier-phase observable measurements in high-precision navigation applications. One intriguing question arises here, if operators of separate constellations are obliged to let some of their signals share the same carrier frequency in order to achieve interoperability, then what about RFC, isn't it degraded by this induced increase in the level of spectral-overlapping between the signals which share the same carrier frequency? In other words, is there a trade-off here between interoperability and compatibility (RFC to be exact)? Unfortunately, the answer is yes, there is a trade-off between interoperability and RFC. Particularly, enhanced interoperability comes at the cost of degraded RFC. In [52], it was shown that a total number of 70 satellites which transmit spectrally overlapping DS-CDMA signals is the best compromise in terms of increasing signal availability while keeping MAI-A/R at tolerable level. It is to be recalled here, signal availability is related one way or another to interoperability. To be more specific, one possible way to increase signal availability is to adopt multi-constellation positioning, which is in turn facilitated when there is interoperability among involved constellations. It is worthwhile to mention, one of the intended aims for using *Multiplexed Binary Offset Carrier* (*MBOC*) modulation in GNSS signals is to minimize their MAI-R when they share the same carrier frequency with BPSK-R signals ([24] Ch. 3, pp. 49-50). In fact, MBOC is a relatively new modulation scheme developed in a collaborative effort by specialists from both US and EU, who had carefully investigated the compatibility and interoperability issues among GPS and Galileo constellations. Subsequently, for the sake of interoperability between GPS and Galileo constellations, a special MBOC implementation known as CBOC(6,1,1/11) was agreed upon to become the basis for the Galileo E1 Open Service signals, which are known to be transmitted using the same carrier frequency of the well-established GPS L1 signals [37]. However, the fact that some constellations deploy MBOC modulation for part of their signals was not forgotten when the optimal number of interoperable satellites (70 as mentioned earlier) was derived.

4.3.2 Coordinate Reference Frame

The engineers, astronomers, and geodesists who are working for different GNSS constellations have developed different *Terrestrial Reference Frame* (*TRF*) realizations of *Earth-Centered and Earth-Fixed* (*ECEF*) coordinate systems, where each realization is developed specifically for an individual constellation. These different TRFs were named

as following: WGS84 for GPS, PZ-90 for GLONASS, CGS2000 for BDS, and GTRF for Galileo. However, these TRFs play direct rule in the determination of the satellite's orbital information, which is in turn incessantly transmitted from the satellite itself down to the user-segment in the form of ephemeris navigation data. Therefore, if two constellations are meant to be interoperable with each other, then any significant inconsistency between their TRFs leads to invalidation of one of the interoperability rules which should have been supposedly enforced upon them. It is worth mentioning, the different GNSS operators had realized that if their constellations are going to be ever interoperable with each other, then they must follow a universal TRF, which is what the operators have agreed upon indeed. Specifically, the universal TRF is known as the *International Terrestrial Reference Frame* (*ITRF*). Although WGS84 and PZ-90 used to initially exhibit up to several meters offset from each other, the operators of the modernized GPS and GLONASS constellations have, respectively, updated WGS84 and PZ-90, thus making them consistent with each other down to the centimeter-level. The purpose of updating both WGS84 and PZ-90 was to align them as much as possible to the latest released version of ITRF, thus increasing their consistency level in the process. Eventually, unlike the case of GPS and GLONASS, the operators of the more recently developed BDS and Galileo constellations have, respectively, enforced their CGS2000 and GTRF to be aligned with the ITRF strictly down to the centimeter-level from the inception of both constellations ([16] Ch. 2).

4.3.3 Time Reference Frame

Accurate absolute timing is an essential component needed for maintaining many of GNSS functionalities. Specifically, it is important to keep the absolute timing accurate down to the sub-nanosecond level within each constellation. Moreover, absolute timing is the key for determining the PVT estimates, especially when it comes to calculating the propagation durations of the signals which have been transmitted all the way from satellites down to user-segments through air interface. However, in order to keep all satellites belonging to each constellation time synchronized perfectly with one another, a time reference frame called *system-time* must be defined and maintained within the constellation. System-times employed by all current GNSS constellations are named as following: GPST for GPS, GLST for GLONASS, BDT for BDS, and GST for Galileo ([16] Ch. 2, pp. 30-31). Furthermore, these system-times are usually imperfectly synchronized with one another. Hence, there are time offsets in the level of nanoseconds between different system-times, for which if not compensated, is going to be detrimental for high-accuracy

multi-constellation positioning applications. In order to synchronize these different system-times together, either one of the following two alternatives must be carried out. First alternative is to determine the time offset between two system-times at the user-segment level, especially in those receivers which are capable of multi-constellation positioning. It is to be noted that, first alternative comes with a penalty, which is an increment of the minimum number of required observation equations (either pseudorange or carrier-phase equations) by one extra equation, so the minimum number of required observation equations in this case becomes five. To make it clear, an entire observation measurement obtained from tracking one of the satellites will be consumed totally for the sake of uncovering the system-time offset which exists between the two constellations of interest to the user-segment. The second alternative, two GNSS operators may choose to increase the level of interoperability between their respective constellations via broadcasting the system-time offset which exists between the two constellations, as part of their respective navigation data broadcast. Currently, the second alternative is implemented between GPS and Galileo constellations as part of the commitment made by their operators of maintaining a satisfactory level of compatibility and interoperability between the two constellations [50].

5. IF-LEVEL SIMULATOR MODEL USING MATLAB/SIMULINK®

5.1 Introduction to the Simulator Model

As was mentioned in *section 1.3*, a complete MATLAB/Simulink® IF-level simulator model for studying Galileo E1, Galileo E5, GPS L1, and GPS L5 radionavigation signals was developed while elaborating this M.Sc. thesis in the Laboratory of Electronics and Communications Engineering at the former Tampere University of Technology (TUT), now Tampere University (TAU). The simulator model includes several wireless transmission channel phenomena, namely: Doppler-shift, multipath-propagation, thermal noise, and wideband/narrowband interference. In this chapter, relevant parts of the simulator model are presented in a rather detailed manner. Nonetheless, for the sake of conciseness, the technical details surrounding MATLAB/Simulink® implementations as well as operating instructions are for the most part omitted. Thus, in a strict sense, this chapter cannot serve as a user guide (or instruction manual) for the simulator model. As seen in *Figure 5.1*, the simulator model consists of three major line-connected units per supported radionavigation signal, and they are namely: transmitting unit, transmission channel unit, and receiving unit. Each one of these units is covered in detail in the continuation of this chapter. From Simulink® perspective, the three aforementioned units contain several modules (or subsystems) inside them, which are in turn made of smaller and smaller building blocks. It is worthwhile to mention, the simulator model contains a switching unit (which is the large yellow block visible in *Figure 5.1*), whose job is to perform a superposition of a certain combination of the four supported radionavigation signals according to the desire of the end-user. In addition, the simulator model contains a *Graphical User-Interface (GUI)* dialog box, whose job is to gather a handful of user-input parameters from the end-user (for more details refer to *Subsection 5.5.1*). Not to mention, there is also a GUI panel, whose job is to give the end-user a visualization of various relevant end results (for more details refer to *Subsection 5.5.2*).

Throughout this chapter, the simulator model is assumed to be operating in regular simulation mode, which is different from the statistical simulation mode that is covered in the next chapter. For example, while the regular simulation mode cannot operate unless MATLAB/Simulink® is running inside a desktop environment which supports graphical shells (e.g., Microsoft Windows 10), the statistical simulation mode can fully operate

within a MATLAB/Simulink® that is running inside a user-interface environment which supports *Command-Line Interface* (*CLI*) shells exclusively. The reason behind why statistical simulation mode can operate within any user-interface environment, whether it supports graphical shells or not, is given as: the statistical simulation mode is designed to use neither the GUI dialog box for gathering user-input parameters nor the GUI panel for visualization of relevant end results (for more details refer to *section 6.1*). However, operating the simulator model in the regular simulation mode can be summarized as following. Firstly, the end-user launches the simulator model, which is composed of several files and folders, whose names are all saved within a Simulink® project package, which is in turn contained within a file with .prj extension. Secondly, a simulation is initiated by the end-user once he/she executes the .m script file responsible for launching simulations in the regular mode. Thirdly, the GUI dialog box for gathering user-input parameters pops up to the end-user asking him/her to supply desired values for the prompted user-input parameters. Fourthly, after the end-user has supplied all the desired values for the prompted user-input parameters, he/she presses the “proceed” button. After that, execution of the simulation starts immediately, and in the meantime, the end-user has to wait patiently until the simulation completes its execution. It is to be noted, the time duration elapsed while a simulation is executing from start to end (i.e., simulation processing time, which is denoted herein as T_{sim}) depends on the values of the user-input parameters which were supplied by the end-user. In addition, the simulation processing time T_{sim} depends also on the hardware capabilities of the underlying computer. Finally, if nothing has interrupted the simulation execution, the GUI panel for visualization of relevant end results pops up immediately after the successful termination of the simulation.

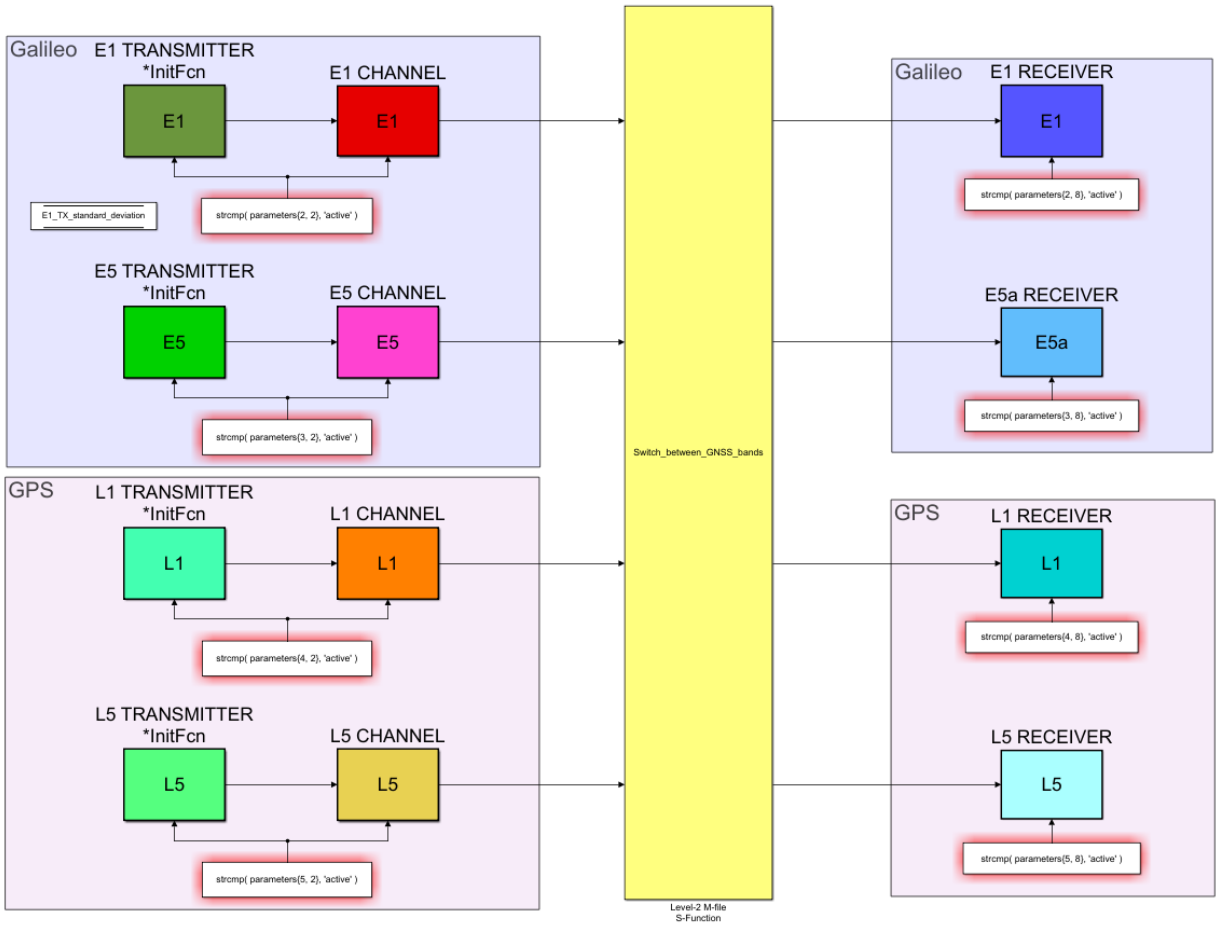

Figure 5.1: A Simulink® diagram which depicts the top-level hierarchy in the simulator model.

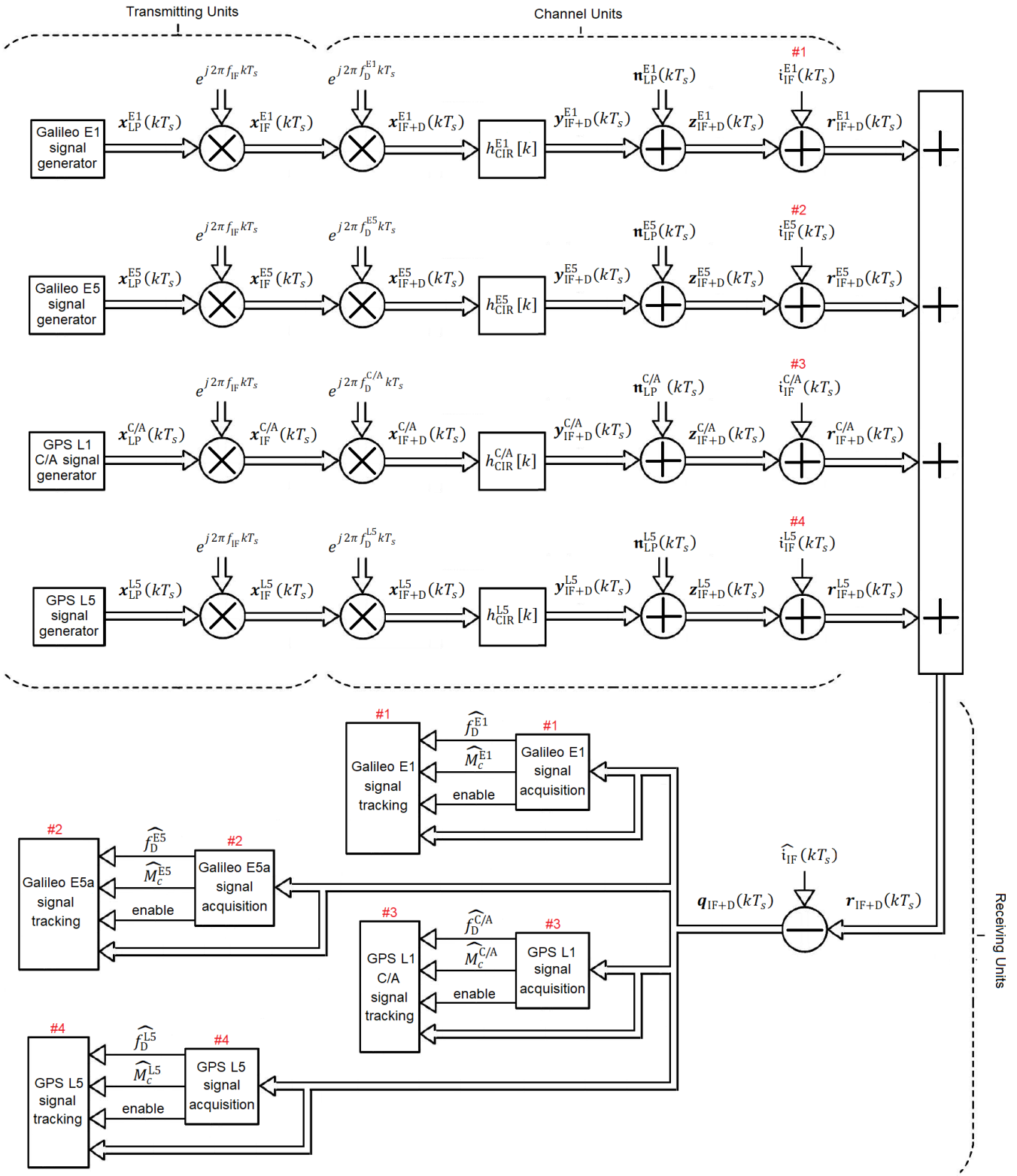

Figure 5.2: A block diagram illustrating the main components included in each major unit in the simulator model. Most of the components of this figure are demystified throughout chapter 5 of this M.Sc. thesis. It is to be noted that each group of components which are tagged with the same red hash number are mutually exclusive with other groups of components which are tagged with different red hash numbers. In other words, when any of the components of a certain group is activated in a given simulation, then all components of every other group must be deactivated in that simulation.

5.2 Transmitting Unit

5.2.1 Generation of Navigation Data

The simulator model in question is dedicated solely for studying different aspects of GNSS signal-processing. Hence, the receiver's navigation-processor stage, which takes care of the navigation data's logical-bits after they have been demodulated successfully, is not implemented anywhere in the simulator model. In addition, from the standpoint of a receiver's signal-processor stage (which is of utmost importance in this M.Sc. thesis), the navigation data of each simulated signal plays no role other than introducing $180°$ phase shifts (aka, polarity reversals) repeatedly to the generated PRN spreading codes. Therefore, the least of our concerns is to generate a legitimate navigation data which includes both *Almanac* and *Ephemeris*. However, those $180°$ phase shifts could be easily simulated using random data. Consequently, navigation data, which is a constituent component of every supported radionavigation composite signal, is simply generated using uniformly-distributed and binary-valued random numbers (which belong to the alphabet $\{-1, +1\}$). Furthermore, the random navigation data, which is generated at the TX-side (i.e., the segment of the simulator model which contains the transmitting units), is not demodulated (or retrieved) whatsoever in the RX-side (i.e., the segment of the simulator model which contains the receiving units).

5.2.2 Carrier Modulation

5.2.2.1 Brief Overview of General I/Q Modulation Principle

In practical GNSS transmitters, the generated signals which are set to traverse the air interface are all continuous-time, real-valued, and bandpass signals, whose spectral contents are located around designated RF carrier frequencies f_c. All of those characteristics are also found in the arbitrary bandpass signal $v_{\mathrm{BP}}(t)$ which is presented in *Equation 5.1*. It is worth mentioning that due to the real-valued nature of practical bandpass signals, the underlying spectra always exhibit *Hermitian symmetry*.

$$v_{\mathrm{BP}}(t) = a(t)\cos\big(2\pi f_c t + \phi(t)\big) \qquad 5.1$$

Where:

- $a(t)$ denotes the instantaneous amplitude of an arbitrary bandpass signal.
- $\phi(t)$ denotes the instantaneous phase of the arbitrary bandpass signal.

Interestingly, such as presented in *Equation 5.2*, using some trigonometric identities, the arbitrary bandpass signal $v_{\mathrm{BP}}(t)$ which was presented in *Equation 5.1* could be represented differently in a more informative way.

$$v_{\mathrm{BP}}(t) = v_{\mathrm{I}}(t)\cos(2\pi f_c t) - v_{\mathrm{Q}}(t)\sin(2\pi f_c t) \qquad 5.2$$

Where:

- $v_{\mathrm{I}}(t) = a(t)\cos\big(\phi(t)\big)$ denotes in-phase component of a bandpass signal.
- $v_{\mathrm{Q}}(t) = a(t)\sin\big(\phi(t)\big)$ denotes quadrature component of the bandpass signal.

The so-called complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ of the arbitrary bandpass signal $v_{\mathrm{BP}}(t)$, which was presented in *Equation 5.2*, is presented in *Equation 5.3*.

$$\boldsymbol{v}_{\mathrm{LP}}(t) = v_{\mathrm{I}}(t) + jv_{\mathrm{Q}}(t) \qquad 5.3$$

It is to be noted, the spectrum (which is usually band-limited) for any complex-valued lowpass-equivalent signal is the same as the zero-centered contents of the positive frequency-axis of the spectrum of its bandpass version. An important formula is the one upconverting a complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ into a real-valued bandpass signal $v_{\mathrm{BP}}(t)$ with the help of a complex-valued exponential oscillator, such as presented in *Equation 5.4*.

$$v_{\mathrm{BP}}(t) = \mathrm{Re}\big\{\, \boldsymbol{v}_{\mathrm{LP}}(t)\, e^{j2\pi f_c t} \,\big\} \qquad 5.4$$

The bandpass signal $v_{\mathrm{BP}}(t)$ generated from *Equation 5.4* is both real-valued and continuous-time (or more precisely analog), and this holds true even in case $\boldsymbol{v}_{\mathrm{LP}}(t)$ was complex-valued and/or discrete-time (or even digital). The importance of *Equation 5.4* is evident from the fact that it is possible to let both components of the complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ (i.e., the in-phase and the quadrature) to be completely independent of each other while they are being modulated into the same carrier frequency, thus saving valuable spectrum (or bandwidth) in the process. What is even more interesting, is that, by using so-called *interplexing*, which is a fairly complicated multi-

plexing technique, then any three independent BPSK-R signals can be combined together on the same carrier frequency while maintaining a constant amplitude through special exploitation of both in-phase and quadrature components of the carrier signal ([13] Ch. 4, pp. 115-116). However, at first glance, *Equation 5.4* seems computationally inefficient because it gives the misleading impression that the imaginary part has been purposelessly computed and then discarded. But the fact is, in practical systems, the imaginary part of *Equation 5.4* is never computed in the first place. There is another important formula, which downconverts the real-valued bandpass arbitrary signal $v_{\mathrm{BP}}(t)$ into the complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ using a conjugated complex-valued exponential oscillator, as presented in *Equation 5.5*.

$$\boldsymbol{v}_{\mathrm{LP}}(t) + \boldsymbol{v}^{*}_{\mathrm{LP}}(t)\, e^{-j4\pi f_c t} = v_{\mathrm{BP}}(t)\, e^{-j2\pi f_c t} \qquad 5.5$$

Where:
- $\boldsymbol{v}^{*}_{\mathrm{LP}}(t)$ denotes a conjugated version of the complex-valued $\boldsymbol{v}_{\mathrm{LP}}(t)$.

One could easily spot from *Equation 5.5* that the left-hand side of the equation contains indeed the desired complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$. However, the desired $\boldsymbol{v}_{\mathrm{LP}}(t)$ in *Equation 5.5* is accompanied by an undesired high-frequency term $\boldsymbol{v}^{*}_{\mathrm{LP}}(t)\, e^{-j4\pi f_c t}$. Therefore, *Equation 5.5* cannot be used alone to downconvert the real-valued bandpass signal $v_{\mathrm{BP}}(t)$ into the complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$. Nonetheless, if the desired complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ is to be isolated from the accompanying undesired high-frequency term found in the left-hand side of *Equation 5.5*, then a *Low-Pass Filter* (*LPF*) should be used for this purpose. In fact, any fine-tuned LPF can easily take care of (i.e., suppress) the high-frequency term found in the left-hand side of *Equation 5.5*, which leaves the desired $\boldsymbol{v}_{\mathrm{LP}}(t)$ solitary in the process. Although *Equation 5.5* is of no use in practical super-heterodyne receivers (for more details refer to *Subsection 5.4.1*), it is, nevertheless, still relevant for some of our upcoming discussions.

5.2.2.2 Software Implementation Considerations

The I/Q modulation which was overviewed previously in *Subsection 5.2.2.1*, is used extensively in digital telecommunication systems, GNSS signals, and some legacy analog systems as well, thereby its importance cannot be overestimated. However, I/Q modulation for the simulator model's TX-signals (i.e., the signals propagating out of the transmitting units) is not carried out exactly as in practical systems. In the continuation of this

subsection, the reason behind why the I/Q modulation which is implemented in the simulator model differs from what is found in practical systems, is going to be explained thoroughly using three distinct arguments.

The first discrepancy between the simulator model and practical systems in terms of carrier modulation is that, throughout the simulator model, the frequencies allocated for the carrier signals are not equivalent to their designated L-band carrier frequencies (which are found in *Figure 3.6*) – instead, every carrier signal in the simulator model is always allocated a frequency which corresponds to IF (i.e., $f_c = f_{\mathrm{IF}}$ throughout the simulator model), and this holds true even at TX-side. Although the fact that all the simulator model's TX-signals are carrier modulated into f_{IF} does not seem logical at a first glance, it is, nevertheless, justified in the following argument. Since receivers in the RX-side are assumed to be super-heterodyne, RX-signals will be downconverted to IF anyway at the RX-side, and this holds true regardless of what carrier frequencies were used in the TX-side. In addition, some modules inside the receivers (signal acquisition and tracking modules to be exact) start their operations first by performing a wipe off of residual digital carrier components found in the RX-signals (i.e., digital IF carrier + Doppler-shift) – in other words, some modules begin their work by digitally basebanding the RX-signal. Therefore, regardless of into which carrier frequency f_c the TX-signal is modulated, it bears no appreciable effect on the receiver's signal-processor stage which is of utmost importance in this M.Sc. thesis. Second justification, according to *Nyquist–Shannon sampling theorem*, which is stated as following: "if an arbitrary signal $v(t)$ contains no frequencies higher than B_{w}, then $v(t)$ is determined completely by giving its ordinates at a series of samples spaced by $1/2B_{\mathrm{w}}$ seconds apart of each other" [53]. Consequently, simulating a carrier frequency f_c which falls into the L-band, requires a sampling-rate R_s whose value is somewhere between (2×10^9) and (4×10^9) $[\mathrm{sample/s}]$. This range of sampling-rates is extremely high, and any existing general-purpose computer is guaranteed to fail coping with it for any reasonable amount of simulation time. In short, it is computationally infeasible for today's general-purpose computers, even the high-end ones, to simulate signals which are modulated into carrier frequencies which fall into the L-band, and this holds true apart from rare and very costly supercomputers.

The second discrepancy between the simulator model and practical systems in terms of carrier modulation is that, the GNSS bandpass signals which traverse the air interface between the satellites' transmitting antennas and the user-segment's receiving antenna

are all continuous-time, or more precisely analog (i.e., continuous in both time and amplitude) – conversely, all signals in the simulator model are discrete-time, or more precisely digital (i.e., discrete in both time and amplitude) from the moment of generation to the moment of termination. Needless to say, microprocessors of existing general-purpose computers handle data which conform strictly to binary-logic; hence, any software running on a general-purpose computer (e.g., MATLAB/Simulink®) is capable of processing either an inherently digital signal or an A/D-converted signal (i.e., a sampled and quantized version of an analog signal). Therefore, as long as general-purpose computers are used, bridging this second discrepancy is impossible to achieve. But fortunately enough, most of what is encountered by GNSS radionavigation signals when they are in the analog-domain (i.e., when they are traversing the air interface), such as: Doppler-shift, multipath-propagation, thermal noise, and interference, is still simulatable in computer-based mathematical models. Generally speaking, the resemblance of those mathematical models to the real-world circumstances has been experimentally and theoretically proven to be fairly accurate.

Third discrepancy between the simulator model and the practical systems in terms of carrier modulation is that, the GNSS bandpass signals which traverse the air interface between the satellites' transmitting antennas and the user-segment's receiving antenna are all real-valued – conversely, the simulator model's TX-signals (which are always carrier modulated using complex-valued exponential IF oscillators) are all complex-valued analytic signals (i.e., imaginary part is computed and preserved throughout). The reason behind why the imaginary parts of simulator model's TX-signals are not discarded as instructed by *Equation 5.4*, is that, preserving the imaginary parts of TX-signals helps the RX-side to operate properly without using any sort of LPF, which should result into a simpler design for the RX-side. It is to be recalled, if the complex-valued $\boldsymbol{v}_{\mathrm{LP}}(t)$ is to be retrieved from a real-valued $v_{\mathrm{BP}}(t)$, then the real-valued $v_{\mathrm{BP}}(t)$ must be mixed with a conjugated complex-valued exponential oscillator; subsequently, in order to remove the undesired high-frequency term which is found in the left-hand side of *Equation 5.5*, the resulting product must pass through a LPF. On the other hand, if the complex-valued $\boldsymbol{v}_{\mathrm{LP}}(t)$ is to be retrieved from a complex-valued analytic signal $\boldsymbol{v}_{\mathrm{BP}}(t)$, then mixing the complex-valued analytic signal $\boldsymbol{v}_{\mathrm{BP}}(t)$ with the conjugated complex-valued exponential oscillator (without being succeeded by any sort of LPF) is sufficient to exclusively retrieve the complex-valued $\boldsymbol{v}_{\mathrm{LP}}(t)$ without the accompanying high-frequency term. However, there is a price to be paid for preserving the imaginary part of TX-signals, which besides slowing down the simulation, consuming more memory is inevitable. As a matter of fact,

complex-valued digital signals consume twice as much memory as real-valued digital signals do.

When *Equation 5.4* is reevaluated after taking the three aforementioned discrepancies into account, we arrive at *Equation 5.6*, which is more realistic in terms of software implementations. In fact, *Equation 5.6*, accurately describes the actual carrier modulation process which is implemented in every TX-signal found in the simulator model.

$$\boldsymbol{v}_{\mathrm{IF}}(kT_s) = \boldsymbol{v}_{\mathrm{LP}}(kT_s)\, e^{j2\pi f_{\mathrm{IF}} kT_s};\ \ k = 1, 2, \ldots \qquad 5.6$$

Where:
- T_s denotes the fundamental sampling-period, which is reciprocal to the fundamental sampling-rate R_s *(i.e.,* $T_s = 1/R_s$*).*
- $e^{j2\pi f_{\mathrm{IF}} kT_s}$ denotes a discrete-time and complex-valued exponential oscillator, whose frequency is set to f_{IF}.
- $\boldsymbol{v}_{\mathrm{LP}}(kT_s)$ denotes an arbitrary discrete-time and usually complex-valued lowpass-equivalent TX-signal.
- $\boldsymbol{v}_{\mathrm{IF}}(kT_s)$ denotes an IF-centered analytic version of $\boldsymbol{v}_{\mathrm{LP}}(kT_s)$.

5.2.3 TX-signals

5.2.3.1 Galileo E1-B/C Composite Signal

As mentioned earlier in *section 1.3*, the simulator model in question, was not built from scratch. Instead, before the author started developing the simulator model, parts of it were developed already by some preceding M.Sc. students. Since Galileo E1 transmitter was largely adopted from an earlier version of the simulator model, and for the sake of conciseness as well, the technical details surrounding the Galileo E1 transmitter are for the most part omitted. However, for those who are interested, the Galileo E1 transmitter which is implemented in the simulator model is described in depth in [54, 55] in addition to [32]. Nonetheless, the simulator model's Simulink® block diagram of the CBOC(6,1,1/11)-modulated and IF-centered Galileo E1-B/C composite TX-signal, is still depicted in *Figure 5.3*.

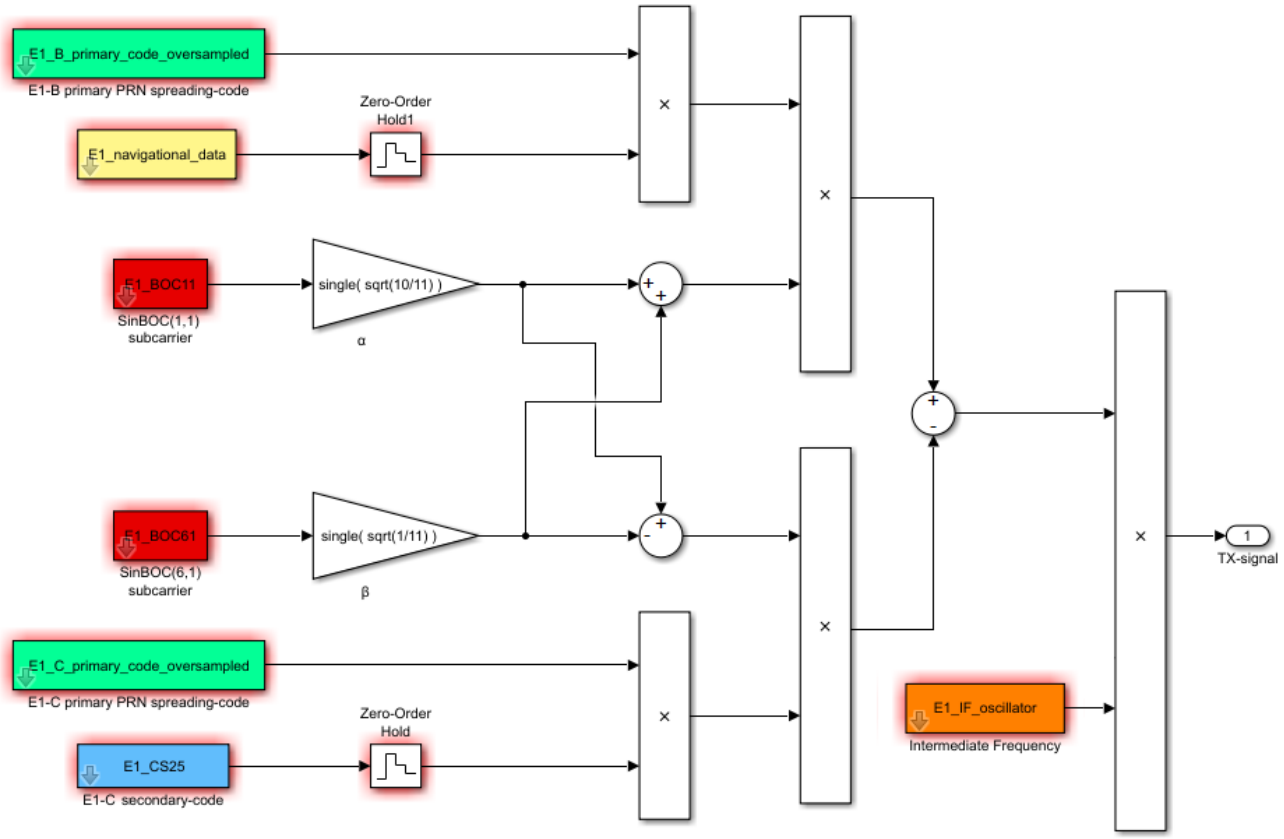

Figure 5.3: Simulink® block diagram of CBOC(6,1,1/11)-modulated and IF-centered Galileo E1-B/C composite TX-signal.

5.2.3.2 Galileo E5a/b Composite Signal

Similarly with the case of Galileo E1 transmitter, the Galileo E5 transmitter was largely adopted from an earlier version of the simulator model. Therefore, for the sake of conciseness, the technical details surrounding the Galileo E5 transmitter are for the most part omitted. However, for those who are interested, the Galileo E5 transmitter which is implemented in the simulator model is described in depth in [55, 56] in addition to [32]. Nonetheless, the simulator model's Simulink® block diagram of the AltBOC(15,10)-modulated and IF-centered Galileo E5a/b composite TX-signal, is still depicted in *Figure 5.4*.

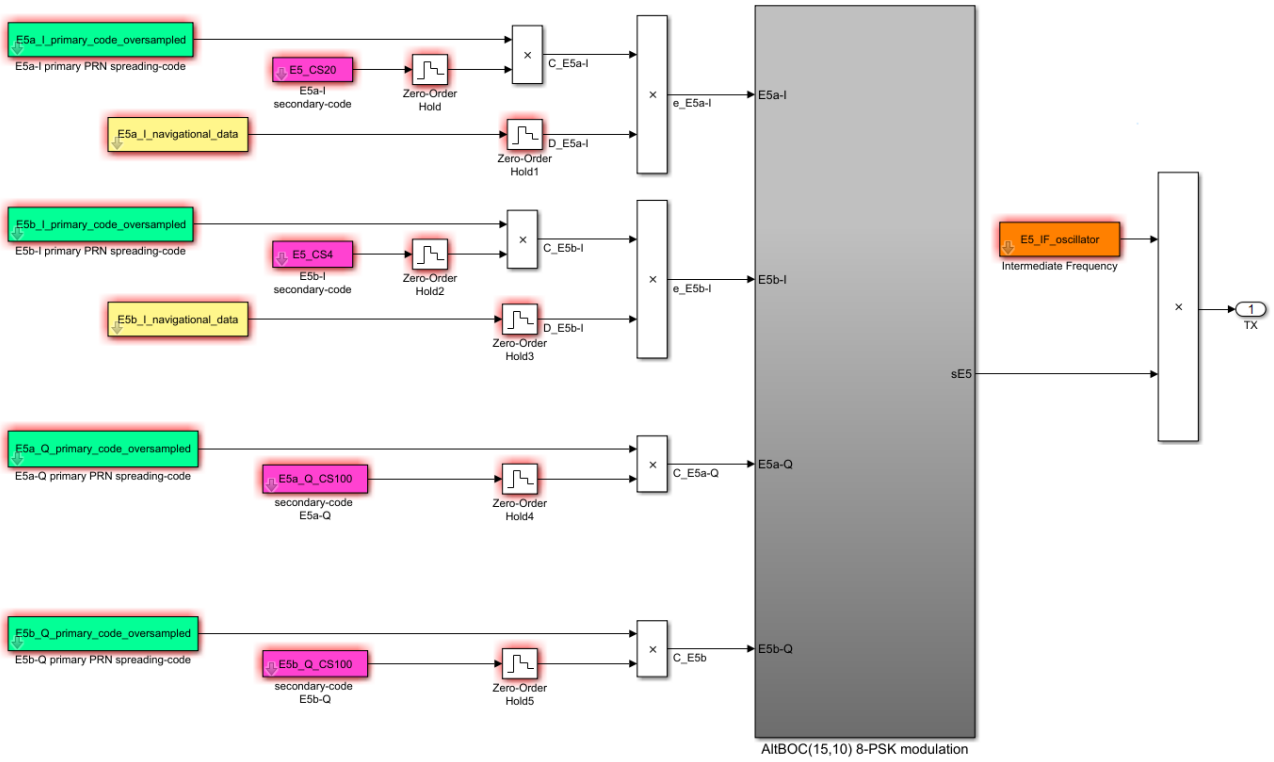

Figure 5.4: Simulink® block diagram of the AltBOC(15,10)-modulated and IF-centered Galileo E5a/b composite TX-signal.

5.2.3.3 GPS L1 C/A Signal

The formula for the (BPSK-R)-modulated GPS L1 bandpass signal, whose time-domain representation was given in ([6] Ch. 9, pp. 347-351), is as presented in *Equation 5.7*.

$$x_{\mathrm{L1}}(t) = \sqrt{2P_{\mathrm{C/A}}}\, d^{\mathrm{L1}}(t)\, g^{\mathrm{C/A}}(t) \cos(2\pi f_{\mathrm{L1}} t + \theta_{\mathrm{L1}}) + \sqrt{2P_{\mathrm{P(Y)}}}\, d^{\mathrm{L1}}(t)\, g^{\mathrm{P(Y)}}(t) \sin(2\pi f_{\mathrm{L1}} t + \theta_{\mathrm{L1}}) \qquad 5.7$$

Where:

- θ_{L1} denotes nominal (but ambiguous) carrier phase-shift.
- f_{L1} denotes carrier frequency for GPS L1 signal, which is $1575.42\ [\mathrm{MHz}]$.
- $g^{\mathrm{C/A}}(t)$ and $g^{\mathrm{P(Y)}}(t)$ denote baseband PAM waveforms of SPS signal's PRN spreading code C/A and PPS signal's PRN spreading code P(Y), respectively.
- $d^{\mathrm{L1}}(t)$ denotes baseband PAM waveform of the navigation data, which is common for both the C/A and the P(Y) signals.
- $\sqrt{2P_{\mathrm{C/A}}}$ and $\sqrt{2P_{\mathrm{P(Y)}}}$ denote amplitude coefficients for the C/A signal and the P(Y) signal, respectively.

Equation 5.7 describes a real-valued, continuous-time, and RF bandpass signal, which is all set to traverse the air interface between the satellites' transmitting antennas and the user-segment's receiving antenna. However, as mentioned earlier in *Subsection 5.2.2.2*, *Equation 5.7* is difficult to implement in software, or even impossible considering its continuous-time nature. It is to be recalled that, the encrypted P(Y) channel, which is a constituent part of the GPS L1 SPS/PPS composite signal, is not publicly accessible (for more details refer to *Subsection 2.1.2* as well as *Table 3.3*). Hence, only C/A channel, to which civilian users have access, is implemented in the simulator model. When the real-valued lowpass-equivalent C/A channel $x_{\mathrm{LP}}^{\mathrm{C/A}}(t)$ is extracted from $x_{\mathrm{L1}}(t)$ found in *Equation 5.9* and subsequently substituted for $\boldsymbol{v}_{\mathrm{LP}}(kT_s)$ found in *Equation 5.6*, then the result is a discrete-time, complex-valued, and IF-centered GPS L1 C/A analytic TX-signal $\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{C/A}}(kT_s)$, which represents what is implemented actually in the simulator model. After assuming $\theta_{\mathrm{L1}} = 0^{\circ}$ also $\sqrt{2P_{\mathrm{C/A}}}$ is normalized, then the resulting $\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{C/A}}(kT_s)$ is as given in *Equation5.8*, and its equivalent Simulink® block diagram is as shown in *Figure 5.5*.

$$\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{C/A}}(kT_s) = \left(d^{\mathrm{L1}}(kT_s)\, g^{\mathrm{C/A}}(kT_s)\right) e^{j2\pi f_{\mathrm{IF}} kT_s}; \quad k = 1, 2, \ldots \qquad 5.8$$

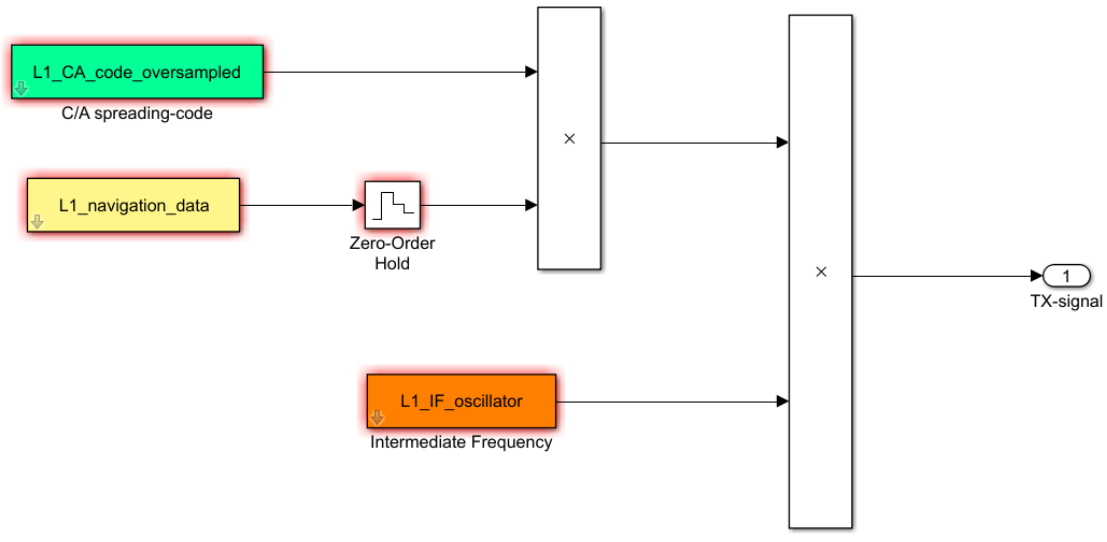

Figure 5.5: Simulink® block diagram of BPSK-R-modulated and IF-centered GPS L1 C/A TX-signal.

5.2.3.4 GPS L5-I/Q Composite Signal

The formula for the QPSK-modulated GPS L5-I/Q composite bandpass signal, whose time-domain representation was given in ([8] Ch. 9, pp. 381-383), is as presented in *Equation 5.9*.

$$x_{\mathrm{L5}}(t) = \sqrt{2P_{\mathrm{L5\text{-}I}}}\left(d^{\mathrm{L5\text{-}I}}(t)\, h^{\mathrm{L5\text{-}I}}(t)\, g^{\mathrm{L5\text{-}I}}(t)\right)\cos(2\pi f_{\mathrm{L5}} t + \theta_{\mathrm{L5}}) + \sqrt{2P_{\mathrm{L5\text{-}Q}}}\left(h^{\mathrm{L5\text{-}Q}}(t)\, g^{\mathrm{L5\text{-}Q}}(t)\right)\sin(2\pi f_{\mathrm{L5}} t + \theta_{\mathrm{L5}}) \quad 5.9$$

Where:

- θ_{L5} denotes nominal (but ambiguous) carrier phase-shift.
- f_{L5} denotes carrier frequency for the GPS L5 signal.
- $g^{\mathrm{L5\text{-}I}}(t)$ and $g^{\mathrm{L5\text{-}Q}}(t)$ denote baseband PAM waveforms of the primary PRN Spreading codes for the L5-I channel and the L5-Q channel, respectively.
- $h^{\mathrm{L5\text{-}I}}(t)$ and $h^{\mathrm{L5\text{-}Q}}(t)$ denote baseband PAM waveforms of the L5-I channel's length 10 Neumann–Hoffman secondary code and the L5-Q channel's length 20 Neumann–Hoffman secondary code, respectively.
- $d^{\mathrm{L5\text{-}I}}(t)$ denotes baseband PAM waveform of the navigation data, which belongs to L5-I channel exclusively.
- $\sqrt{2P_{\mathrm{L5\text{-}I}}}$ and $\sqrt{2P_{\mathrm{L5\text{-}Q}}}$ denote amplitude coefficients for the L5-I channel and the L5-Q channel, respectively.

Similar to the argument presented in *Subsection 5.2.3.3, Equation 5.9* describes a real-valued, continuous-time, and RF bandpass signal which is all set to traverse the air interface between the satellite's transmitting antenna and the user-segment's receiving antenna. However, *Equation 5.9* is difficult to implement in software, or even impossible considering its continuous-time nature. When the complex-valued lowpass-equivalent $\boldsymbol{x}_{\mathrm{LP}}^{\mathrm{L5}}(t)$ is extracted from $x_{\mathrm{L5}}(t)$ found in *Equation 5.9* and subsequently substituted for $\boldsymbol{v}_{\mathrm{LP}}(kT_s)$ found in *Equation 5.6*, then the result is a discrete-time, complex-valued, and IF-centered GPS L5-I/Q composite analytic TX-signal $\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{L5}}(kT_s)$, which represents what is implemented actually in the simulator model. After assuming $\theta_{\mathrm{L5}} = 0^{\circ}$ also $\sqrt{2P_{\mathrm{L5\text{-}I}}}$ and $\sqrt{2P_{\mathrm{L5\text{-}Q}}}$ are both normalized, then the resulting $\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{L5}}(kT_s)$ is as given in *Equation 5.10*, and its equivalent Simulink® block diagram is as shown in *Figure 5.6*.

$$\boldsymbol{x}_{\mathrm{IF}}^{\mathrm{L5}}(kT_s) = \begin{pmatrix} d^{\mathrm{L5\text{-}I}}(kT_s)\, h^{\mathrm{L5\text{-}I}}(kT_s)\, g^{\mathrm{L5\text{-}I}}(kT_s) \\ +\, j\left(h^{\mathrm{L5\text{-}Q}}(kT_s)\, g^{\mathrm{L5\text{-}Q}}(kT_s)\right) \end{pmatrix} e^{j2\pi f_{\mathrm{IF}} kT_s}; \quad k = 1, 2, \ldots \qquad 5.10$$

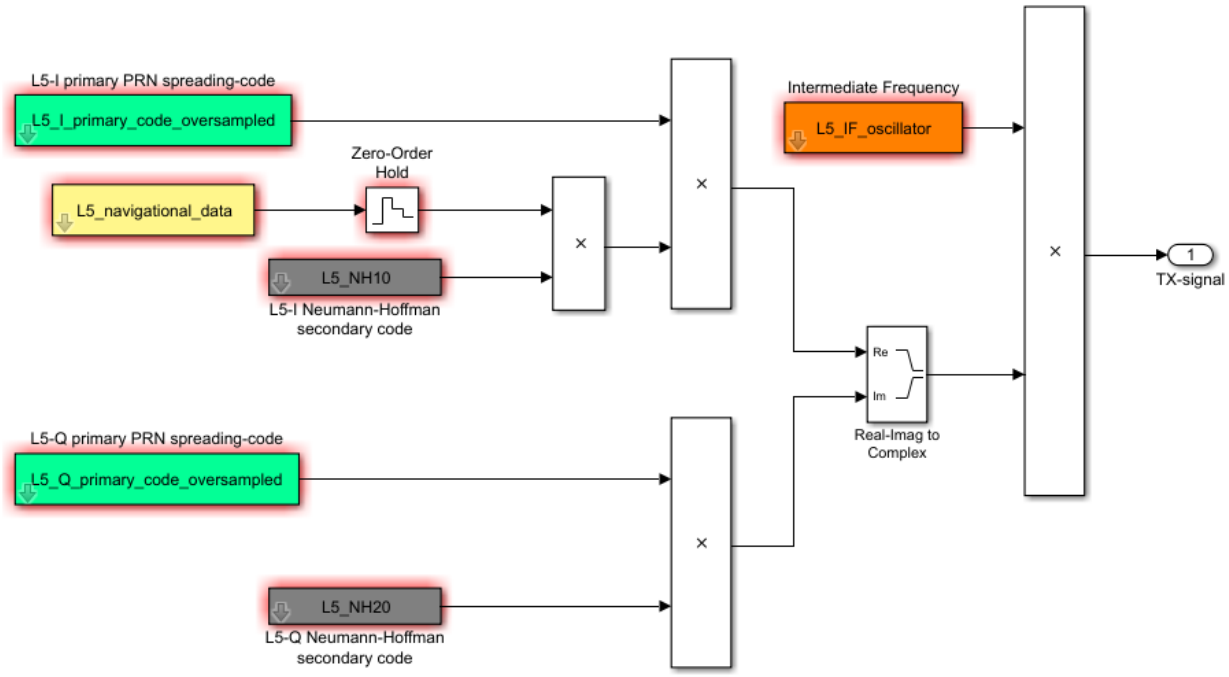

Figure 5.6: Simulink® block diagram of QPSK-modulated and IF-centered GPS L5-I/Q composite TX-signal.

5.3 Transmission Channel Unit

5.3.1 Doppler-Shift

Since GNSS satellites exist in *Medium Earth orbit* (*MEO*) geocentric orbits, they are always in relative motion with respect to their user-segments. Due to this relative motion, Doppler-shift, which affects signals' carrier frequencies f_c, is always present in GNSS radionavigation signals. Doppler-shift itself is quantized as given in *Equation 5.11* ([13] Ch. 2, pp. 58-61).

$$f_{\mathrm{D}} = \frac{\overrightarrow{\Delta v} \cdot \hat{r}}{c} f_c \qquad 5.11$$

Where:
- c denotes the speed of light in vacuum, which is $299\,792\,458\ [\mathrm{m/s}]$.
- $\overrightarrow{\Delta v} \cdot \hat{r}$ denotes a dot product between a satellite to user-segment relative velocity vector and a unit vector pointing from the user-segment towards the satellite.
- f_{D} denotes Doppler-shift frequency with respect to the carrier frequency f_c.

Assuming a maximum line-of-sight relative velocity between a satellite and a stationary user-segment of $800\ [\mathrm{m/s}]$, then the maximum Doppler-shift frequency in lower L-band signals is anticipated to be around $3140\ [\mathrm{KHz}]$ whilst in upper L-band signals it is anticipated to be around $4200\ [\mathrm{KHz}]$ (for details about lower/upper L-band refer to *Figure 3.6*). In the simulator model, frequency of the Doppler-shift f_{D} per activated TX-signal, is a user-input parameter which is supplied directly by the end-user, and thus its value is not computed via *Equation 5.11*. As presented in *Equation 5.12,* Doppler-shift is implemented in the simulator model via mixing the IF-centered TX-signal directly with a complex-valued exponential oscillator whose frequency is set to f_{D}.

$$\boldsymbol{x}_{\mathrm{IF+D}}(kT_s) = \boldsymbol{x}_{\mathrm{IF}}(kT_s)\, e^{j2\pi f_{\mathrm{D}} kT_s}; \quad k = 1, 2, \ldots \qquad 5.12$$

It is to be noted, *Equation 5.12* is applicable to the four supported radionavigation signals in the simulator. However, although f_{D}, $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$, and $\boldsymbol{x}_{\mathrm{IF}}(kT_s)$ of *Equation 5.12* are not marked accordingly, they are, nevertheless, always associated with one of the four supported radionavigation signals in the simulator model whenever *Equation 5.12* is itself associated with that particular radionavigation signal (such as the case of *Figure 5.2*).

5.3.2 Multipath-Propagation

Multipath-propagation phenomenon in GNSS is defined as: if there are scatterers in the air interface between a satellite and its receiving user-segment, then due to reflections and diffractions, the satellite-generated radionavigation signal transits over multiple separate roundabout paths before reaching the user-segment. Multipath-propagation is similar conceptually to the familiar echo phenomenon which is observed usually in sound

waves. But in the GNSS case, multipath-propagation occurs with electromagnetic waves whose spectra are all contained within the L-band, such as depicted in *Figure 5.7*.

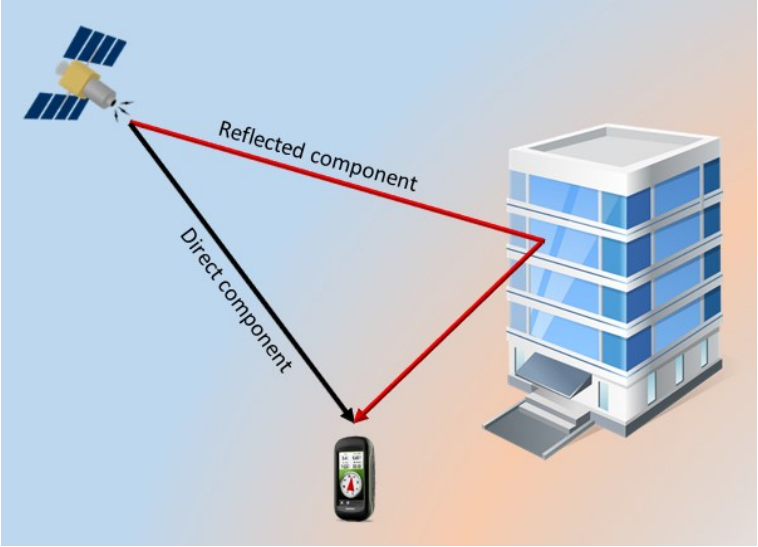

Figure 5.7: Illustration of a simple multipath-propagation scenario comprising a direct (or line-of-sight) signal arriving at the receiver, in addition to a single reflected copy of the direct signal.

Signal $\boldsymbol{y}_{\mathrm{BP}}(t)$ in *Equation 5.13* is a continuous-time, complex-valued, analytic, and bandpass received signal, which is composed of one direct (or line-of-sight) component plus a combination of additional $(N_{\mathrm{mp}}-1)$ reflected (or indirect) components. All the constituent components of $\boldsymbol{y}_{\mathrm{BP}}(t)$, whether they are direct or reflected, are made of a continuous-time, complex-valued, and lowpass-equivalent arbitrary transmitted signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ which has been carrier-modulated by a complex-valued exponential oscillator whose nominal phase-shift is θ_0. However, while the constituent direct component in $\boldsymbol{y}_{\mathrm{BP}}(t)$ was time-delayed by τ_0 and magnitude-scaled by α_0, each one of the constituent reflected components was time-delayed by its corresponding τ_n and magnitude-scaled by its corresponding α_n. It is to be noted that for the sake of simplicity, time-invariance was assumed for *Equation 5.135.13*. However, *Multipath-to-Direct magnitude-Ratio* (*MDR*) $\widetilde{\alpha_n}$, which is given as $\widetilde{\alpha_n} = \alpha_n/\alpha_0$, is the n^{th} reflected component's received magnitude α_n relative to the direct component's received magnitude α_0. Likewise, excess delay $\widetilde{\tau_n}$, which is given as $\widetilde{\tau_n} = \tau_n - \tau_0$, is the n^{th} reflected component's propagation delay τ_n relative to the direct component's propagation delay τ_0. It is worthwhile to mention, the so-called multipath transmission channel's *Power-Delay Profile* (*PDP*) is portrayed graphically by plotting the $(N_{\mathrm{mp}}-1)$ reflected components' taps (or points) whose

abscissas are equivalent to the excess delays $\widetilde{\tau_n}$ and whose ordinates are equivalent to the squared MDRs $(\widetilde{\alpha_n})^2$, that is for all significant reflected components of course ([13] Ch. 6, pp. 281-285).

$$\boldsymbol{y}_{\mathrm{BP}}(t) = \alpha_0\, \boldsymbol{v}_{\mathrm{LP}}(t-\tau_0)\, e^{j2\pi f_c t}\, e^{-j\varphi_0} + \sum_{n=1}^{N_{\mathrm{mp}}-1} \alpha_n\, \boldsymbol{v}_{\mathrm{LP}}(t-\tau_n)\, e^{j2\pi(f_c+\widetilde{f_n})t}\, e^{-j\varphi_n} \qquad 5.13$$

Where:

- $\widetilde{f_n}$ denotes the n^{th} reflected component's received carrier frequency relative to the direct component's received carrier frequency f_c. It is to be noted, "when relative motion between satellites, scatterers, and receiver is different from relative motion between satellites and receiver, this causes the reflected components to encounter Doppler-shifts which are different from the one encountered by the direct component" ([13] Ch. 6, pp. 282).
- φ_0 denotes the direct component's received carrier phase-shift, which is given as $\varphi_0 = 2\pi f_c \tau_0 + \theta_0$, whereas θ_0 in turn denotes the nominal carrier phase-shift.
- φ_n denotes the n^{th} reflected component's received carrier phase-shift, which is given as $\varphi_n = 2\pi\left(f_c + \widetilde{f_n}\right)\tau_n + \theta_0$.

Perhaps the best way to explain the perturbing effect of multipath-propagation on GNSS signals, is to show an example of the type of distortion incurred by the *Cross-Correlation Function* (*CCF*) between a BPSK-R radionavigation signal and its designated reference-code (aka, receiver-generated replica signal) due to multipath-propagation, such as depicted in *Figure 5.8*. Although there is only one reflected component in the multipath-propagation scenario depicted in *Figure 5.8*, it is, nevertheless, considered a severe case of multipath-propagation which will result into a significant error in the PVT estimates.

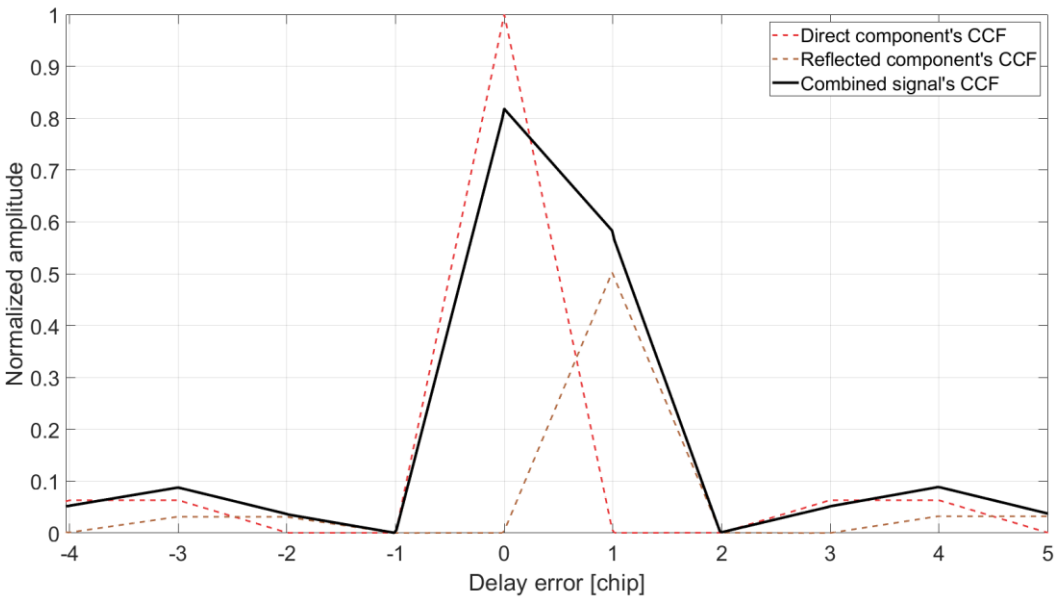

Figure 5.8: Illustration of the distortion inflicted upon the CCF between a BPSK-R radionavigation signal and its designated reference code due to multipath-propation. The tap of the reflected component is given as $((b_1 - b_0) = 1\ [\text{chip}],\quad (\widetilde{\alpha_1})^2 = -3\ [\text{dB}])$.

It is worthwhile to mention few remarks about multipath-propagation implementation in the simulator model. Firstly, θ_0, and $\widetilde{f_n}$ are both assumed zero-valued, whereas f_c is equivalent to f_{IF} (for reasons mentioned in *Subsection 5.2.2.2*). Secondly, the number of taps N_{mp}, which is determined by the simulator model's end-user, cannot exceed five taps (which corresponds to one direct component tap plus four reflected components taps). Thirdly, the n^{th} component's propagation delay τ_n (which includes the direct component's propagation delay τ_0 as well) is defined as the product of multiplying a multiplicand T_c with a multiplier b_n which constitutes the n^{th} component's propagation delay in chips that is supplied directly in $[\text{chip}]$ unit by the end-user. In other words, the propagation delay of the n^{th} component in the simulator model is defined as $\boxed{(\tau_n)_{[\text{s}]} = (b_n)_{[\text{chip}]} \times (T_c)_{[\text{s/chip}]}}$. However, in order to physically delay the n^{th} component in the simulator model according to the quantity of the user-input parameter b_n, then a copy of the pre-multipath signal $\boldsymbol{x}_{\text{IF+D}}(kT_s)$ must be linearly shifted by a number of samples equivalent to M_n, which itself is defined as $\boxed{(M_n)_{[\text{sample}]} = [\![\, (b_n)_{[\text{chip}]} \times (R_s/R_c)_{[\text{sample/chip}]} \,]\!]}$, where $[\![\cdot]\!]$ in turn denotes a function that rounds its argument to the nearest integer, and R_s/R_c is in turn the oversampling factor (i.e., number of samples per chip). Fourthly, the random noise in the simulator model is superimposed over each activated signal dynamically according to a certain predefined C/N_0 parameter (for more details refer to *Subsection 4.3.10*). Hence, signals'

amplitude/power levels themselves are insignificant as long as the correct amplitude/power ratio between any two different radionavigation signals is preserved. In other words, if there was no unfair amplification or attenuation of any signal's amplitude/power with regard to other signals, then the level of any signal's amplitude/power itself does not matter as long as the superimposed noise is scaled dynamically. Even though in practical systems the n^{th} component's received power α_n^2 (which includes the direct component's received power α_0^2 as well) is expressed in either $[\mathrm{W}]$, $[\mathrm{mW}]$, $[\mathrm{dBW}]$, or $[\mathrm{dBm}]$ unit; nevertheless, α_n^2 values in our simulator model are all relative to the power level of the pre-multipath signal $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$ (which is denoted as P_x). Therefore, α_n^2 shall remain unitless when expressed in linear-scale or shall be given $[\mathrm{dB}]$ unit when expressed in decibels. Not to mention, n^{th} component's received power α_n^2 is a user-input parameter that is supplied directly in $[\mathrm{dB}]$ unit by the end-user. After taking the four aforementioned remarks into account, the formula for multipath-propagation, which is implemented actually in the simulator model, is as given in *Equation 5.15*. It is to be noted, in *Equation 5.15*, scaling with the coefficient α, whose value is given in *Equation 5.14*, was carried out in order to ensure that the post-multipath signal $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$ has the same average power of the pre-multipath signal $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$, thus conservation of energy law is satisfied. Since signals' amplitude/power levels themselves are insignificant in the simulator model as mentioned earlier, scaling each active signal with the coefficient α is important only when two or more different radionavigation signals are activated simultaneously. To make it clear, in any given simulation, when multipath-propagation is introduced to any of the activated radionavigation signals, then scaling it with the coefficient α of *Equation 5.14* ensures that the signal's average power is neither amplified nor attenuated unfairly with regard to other activated signals.

$$\alpha = \frac{\sqrt{P_x}}{\sum_{n=0}^{N_{\mathrm{mp}}-1} 10^{(\alpha_n^2)_{[\mathrm{dB}]}/20}} \qquad 5.14$$

$$\boldsymbol{y}_{\mathrm{IF+D}}(kT_s) = \alpha \sum_{n=0}^{N_{\mathrm{mp}}-1} \left(10^{(\alpha_n^2)_{[\mathrm{dB}]}/20}\right) \boldsymbol{x}_{\mathrm{IF+D}}\big((k-M_n)T_s\big); \quad k = 1, 2, \ldots \qquad 5.15$$

Equations 5.14 and *5.15* both accurately describe the way multipath-propagation was implemented in the simulator model; however, a more convenient method for computing

$\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$ is still presented next. The alternative method takes advantage of the fact that multipath-propagation itself is a type of linear distortion system which could be modeled as a *Finite Impulse Response* (*FIR*). In order to be able to present the new method, *Channel Impulse Response* (*CIR*), which is the term usually given for the multipath-propagation's FIR, must be presented first. Assuming time-invariance again, the discrete-time CIR, which is denoted herein as $h_{\mathrm{CIR}}[k]$, is similar somehow to the earlier-mentioned PDP. As presented in *Equation 5.16*, $h_{\mathrm{CIR}}[k]$ is constructed through summation of as many as N_{mp} different amplitude-scaled and time-delayed versions of the discrete-time Dirac delta functions $\delta[k]$. Not to mention, $h_{\mathrm{CIR}}[k]$ is scaled with the coefficient α (whose value is given in *Equation 5.14*) for the same previously mentioned reasons which were applicable to *Equation 5.15*. As presented in *Equation 5.17*, the post-multipath signal $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$ itself is constructed using a convolution operation between $h_{\mathrm{CIR}}[k]$ and pre-multipath signal $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$. Finally, when N_{mp} (whose maximum possible value is 5 as mentioned earlier) is defined in the simulator model to be equivalent to 1, which corresponds to a scenario where no reflected components exist at all, then the post-multipath signal $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$ is basically the same as the pre-multipath signal $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$ with no incurred modifications whatsoever.

$$h_{\mathrm{CIR}}[k] = \alpha \sum_{n=0}^{N_{\mathrm{mp}}-1} \left(10^{(\alpha_n^2)_{[\mathrm{dB}]}/20}\right) \delta[k - M_n]\,; \quad k = 1, 2, \ldots \tag{5.16}$$

$$\begin{aligned} \boldsymbol{y}_{\mathrm{IF+D}}(kT_s) &= h_{\mathrm{CIR}}[k] * \boldsymbol{x}_{\mathrm{IF+D}}(kT_s) \\ &= \sum_{m=0}^{\infty} h_{\mathrm{CIR}}[m]\, \boldsymbol{x}_{\mathrm{IF+D}}\big((k-m)T_s\big)\,; \quad k = 1, 2, \ldots \end{aligned} \tag{5.17}$$

Equations 5.14, *5.15*, *5.16*, and *5.17* are applicable to all of the four supported radionavigation signals in the simulator model. However, although N_{mp}, α, P_x, α_n^2, M_n, $h_{\mathrm{CIR}}[k]$, $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$, and $\boldsymbol{x}_{\mathrm{IF+D}}(kT_s)$ of *Equations 5.14*, *5.15*, *5.16*, and *5.17* are not marked accordingly, they are, nevertheless, always associated with one of the four supported radionavigation signals in the simulator model whenever *Equations 5.14*, *5.15*, *5.16*, and *5.17* are themselves associated with that particular radionavigation signal (such as the case of *Figure 5.2*).

5.3.3 Thermal Noise

Due to random motion of electrons in conducting media, thermal noise (aka, ambient noise) always exists in every telecommunication system, and GNSS is not an exception. In a normal room's temperature, thermal noise is primarily white, i.e., its PSD is entirely constant-valued, which is given as $G_{\mathfrak{n}}(f) = N_0/2\ [\mathrm{W/Hz}]$, and this holds true up to extremely high frequencies (several terahertz in fact). The noise spectral density N_0 itself is computed as given in *Equation 5.18* ([27] Ch. 8, pp. 338-348).

$$(N_0)_{[\mathrm{W/Hz}]} = (K_{\mathrm{B}})_{[\mathrm{W/(\mathcal{K}\ Hz)}]} \times (\mathcal{T})_{[\mathcal{K}]} \qquad 5.18$$

Where:

- $(\mathcal{T})_{[\mathcal{K}]}$ denotes system's noise temperature in $[\mathcal{K}]$ unit, which is the unit for temperature in Kelvin. $\mathcal{T}$ is usually computed using Friis formula for noise temperature in a cascaded system of stages.
- K_{B} denotes Boltzmann constant, which is given as $1.380 \times 10^{-23}\ [\mathrm{W/(\mathcal{K}\ Hz)}]$.

Now let's assume that at the frontend of a certain receiver, thermal noise passes through a real-valued, unity gain, analog, and ideal *Band-Pass Filter* (*BPF*), whose transfer function $H_{\mathrm{BPF}}(f)$ is Hermitian symmetric, strictly rectangular in shape, centered at f_{BPF}, and with a single-sided bandwidth B_{BPF}. Subsequently, due to the filtering process exerted by the frontend BPF, the thermal noise in question becomes band-limited (i.e., no longer white), and hence termed bandpass noise. The PSD of the bandpass noise, which is denoted as $G_{\mathfrak{n}}^{\mathrm{BP}}(f)$, is given in *Equation 5.19*, and depicted in *Figure 5.9*. The bandpass noise's average power, which is denoted by $P_{\mathfrak{n}}$, is linearly proportional to the bandwidth B_{BPF} of the frontend BPF, as given in *Equation 5.20.*

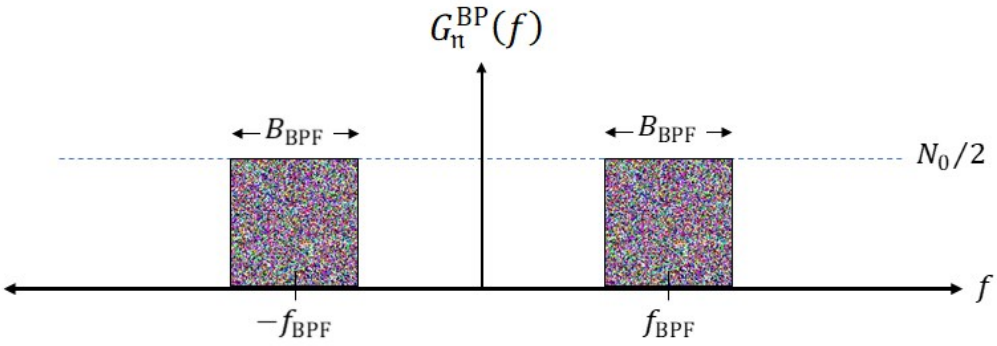

Figure 5.9: Illustration of the PSD of bandpass noise after passing through an ideal BPF.

$$G_{\mathrm{n}}^{\mathrm{BP}}(f) = G_{\mathrm{n}}(f)\,|H_{\mathrm{BPF}}(f)|^2 = \begin{cases} N_0/2, & f_{\mathrm{BPF}} - \dfrac{B_{\mathrm{BPF}}}{2} \le |f| \le f_{\mathrm{BPF}} + \dfrac{B_{\mathrm{BPF}}}{2} \\ 0, & \text{otherwise} \end{cases} \qquad 5.19$$

$$P_{\mathrm{n}} = \int_{-\infty}^{+\infty} G_{\mathrm{n}}^{\mathrm{BP}}(f)\,df = N_0\,B_{\mathrm{BPF}} \qquad 5.20$$

Just like in the case of information-bearing signals (such as the one presented in *Equation 5.2*), the bandpass noise, which is denoted as $\mathrm{n}_{\mathrm{BP}}(t)$, contains both an in-phase component $\mathrm{n}_{\mathrm{I}}(t)$ and a quadrature component $\mathrm{n}_{\mathrm{Q}}(t)$, such as presented in *Equation 5.21*. It is to be noted, both $\mathrm{n}_{\mathrm{I}}(t)$ and $\mathrm{n}_{\mathrm{Q}}(t)$ are stationary Gaussian-distributed noise signals with zero-valued mean (i.e., $\mu_{\mathrm{n}} = 0$), which are always statistically independent of each other as well. In addition, similar to *Equation 5.3*, the lowpass-equivalent for the $\mathrm{n}_{\mathrm{BP}}(t)$, which is denoted as $\mathbf{n}_{\mathrm{LP}}(t)$, is constructed from both $\mathrm{n}_{\mathrm{I}}(t)$ and $\mathrm{n}_{\mathrm{Q}}(t)$, such as presented in *Equation 5.22*. As a matter of fact, $\mu_{\mathrm{n}} = \mathrm{E}\{\,\mathrm{n}_{\mathrm{I}}(t)\,\} = \mathrm{E}\{\,\mathrm{n}_{\mathrm{Q}}(t)\,\} = \mathrm{E}\{\,\mathrm{n}_{\mathrm{I}}(t)\,\mathrm{n}_{\mathrm{Q}}(t)\,\} = 0$; therefore, the variance σ_{n}^2 of the bandpass noise $\mathrm{n}_{\mathrm{BP}}(t)$, which is synonymous with its average power P_{n}, is as given in *Equation 5.23*.

$$\mathfrak{n}_{\mathrm{BP}}(t) = \mathfrak{n}_{\mathrm{I}}(t)\cos(2\pi f_{\mathrm{BPF}}t) - \mathfrak{n}_{\mathrm{Q}}(t)\sin(2\pi f_{\mathrm{BPF}}t) \qquad 5.21$$

$$\boldsymbol{\mathfrak{n}}_{\mathrm{LP}}(t) = \mathfrak{n}_{\mathrm{I}}(t) + j\mathfrak{n}_{\mathrm{Q}}(t) \qquad 5.22$$

$$\sigma_{\mathfrak{n}}^2 = P_{\mathfrak{n}} = \mathrm{E}\{\,\mathfrak{n}_{\mathrm{I}}^2(t)\,\} = \mathrm{E}\{\,\mathfrak{n}_{\mathrm{Q}}^2(t)\,\} = \mathrm{E}\{\,\mathfrak{n}_{\mathrm{BP}}^2(t)\,\} = N_0\,B_{\mathrm{BPF}} \qquad 5.23$$

In GNSS discipline, *carrier-to-noise-density-power-ratio* (denoted as C/N_0), which is usually measured in $[\mathrm{dB\text{-}Hz}]$ unit, is the parameter commonly used instead of the prominent *Signal-to-Noise-power-Ratio* (*SNR*), and this holds true in our simulator model as well. However, if the frontend bandwidth B_{BPF} is known, then C/N_0 could be easily converted to SNR and vice versa. It is to be noted, because the noise spectral density N_0 itself is not evaluated at all in the simulator model, the bandpass noise variance $\sigma_{\mathfrak{n}}^2$ (which is equivalent to the average power $P_{\mathfrak{n}}$) is not computed via substituting the value of N_0 (which is computed using *Equation 5.18* for example) into *Equation 5.23* (or *Equation 5.20*). Instead, as presented in *Equation 5.24*, bandpass noise variance $\sigma_{\mathfrak{n}}^2$ is computed in the simulator model using both the user-input parameter C/N_0 (which itself is supplied directly in $[\mathrm{dB\text{-}Hz}]$ unit by the end-user) and the signal's average power (denoted as P_y), that is with the help of *Equation 5.23* of course.

$$\sigma_{\mathfrak{n}}^2 = \frac{P_y\,B_{\mathrm{BPF}}}{10^{(C/N_0)_{[\mathrm{dB\text{-}Hz}]}/10}} \qquad 5.24$$

In the simulator model, the well-known *Additive White Gaussian Noise* (*AWGN*) model is implemented. Therefore, after evaluating $\sigma_{\mathfrak{n}}^2$ according to *Equation 5.24*, a discrete-time version of $\boldsymbol{\mathfrak{n}}_{\mathrm{LP}}(t)$ (denoted as $\boldsymbol{\mathfrak{n}}_{\mathrm{LP}}(kT_s)$) is generated, and its constituent in-phase and quadrature components are, respectively, given as: $\mathfrak{n}_{\mathrm{I}}(kT_s) \sim \mathcal{N}(\mu_{\mathfrak{n}},\ \sigma_{\mathfrak{n}}^2)$ and $\mathfrak{n}_{\mathrm{Q}}(kT_s) \sim \mathcal{N}(\mu_{\mathfrak{n}},\ \sigma_{\mathfrak{n}}^2)$, where $\mathcal{N}(\mu_{\mathfrak{n}},\ \sigma^2)$ refers in turn to a *Probability Density Function* (*PDF*) of a normal distribution whose mean is $\mu_{\mathfrak{n}}$ (which is always zero-valued as mentioned earlier) and whose variance is $\sigma_{\mathfrak{n}}^2$. Afterwards, in order to perform superimposing of the generated thermal noise, $\boldsymbol{\mathfrak{n}}_{\mathrm{LP}}(kT_s)$ is algebraically added to the pre-noise signal $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$, such as presented in *Equation 5.25*, where $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$ denotes the post-noise signal.

$$\boldsymbol{z}_{\mathrm{IF+D}}(kT_s) = \boldsymbol{y}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{\mathfrak{n}}_{\mathrm{LP}}(kT_s); \quad k = 1, 2, \ldots \qquad 5.25$$

Equations 5.24 and *5.25* are applicable to all of the four supported radionavigation signals in the simulator model. However, although P_y, B_{BPF}, $\sigma_{\mathfrak{n}}^2$, $\boldsymbol{\mathfrak{n}}_{\mathrm{LP}}(kT_s)$, $\boldsymbol{y}_{\mathrm{IF+D}}(kT_s)$, and $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$ of *Equations 5.24* and *5.25* are not marked accordingly, they are, nevertheless, always associated with one of the four supported radionavigation signals in the simulator model whenever *Equations 5.24* and *5.25* are themselves associated with that particular radionavigation signal (such as the case of *Figure 5.2*).

In the simulator model, there are neither analog nor digital filters of any sort implemented, which is a shortcoming obviously, especially when it comes to the Galileo E5 radionavigation signal. Nonetheless, when Galileo E5 TX-signal is activated, then the total lack of filters in the simulator model is compensated for using a simple workaround solution, which is presented thoroughly in the continuation of this paragraph. It is to be noted, for the sake of calculating the noise variance $\sigma_{\mathfrak{n}}^2$ using *Equation 5.24*, the bandwidth B_{BPF} of the frontend BPF, which does not exist physically in the simulator model, must be supplied by the end-user after he/she assumes its value according to certain bandwidths designated typically for the activated radionavigation signals. However, before discussing the workaround solution, it is to be recalled that the AltBOC(15,10)-modulated Galileo E5a/b composite radionavigation signal, which is referred to shortly as full-band E5 signal, includes both an E5a lower-sideband and an E5b upper-sideband. Dissimilar to other simulator model's receivers which process the entirety of their signals, the simulator model's Galileo E5a receiver processes only the E5a lower-sideband part from the full-band E5 signal. While intentionally neglecting the E5b upper-sideband part, the simulator model's Galileo E5a receiver processes the E5a lower-sideband using similar procedures to those dedicated for QPSK-modulated signals. However, even though E5b upper-sideband continues to exist throughout the simulator model's receiver stages; nevertheless, we can still virtually imitate the effect left by an ideal frontend analog BPF which is designed to completely filter out (i.e., suppress) the E5b upper-sideband part from the full-band E5 signal, that is without touching the E5b upper-sideband part physically (i.e., leaving it unfiltered throughout). The virtual imitation of the ideal frontend BPF works simply via excluding both the average power and the bandwidth of E5b upper-sideband exclusively from the calculations associated with the noise variance $\sigma_{\mathfrak{n}}^2$. Therefore, before proceeding to the stage where the noise variance $\sigma_{\mathfrak{n}}^2$ is calculated using

Equation 5.24, one should artificially tamper with the value of the full-band E5 signal's average power P_y, preferably via dividing it over a factor of 2. Needless to say, when amplitude quantities are concerned, the dividing factor is $\sqrt{2}$ instead. Furthermore, the limits for the bandwidth B_{BPF} of the ideal frontend BPF (which again, does not exist physically in the simulator model) must extend just enough to include the E5a lower-sideband only. To conclude this subsection, once the ideal frontend BPF is virtually imitated according to what has been presented in this paragraph, then the fact that E5b upper-sideband continues to exist throughout the Galileo E5a receiver, which is designed to process the E5a lower-sideband only, is not harmful in itself from signal acquisition and tracking perspective apart from reducing the computational efficiency of the simulator model.

5.3.4 Interference

5.3.4.1 Narrowband Interference

The types of interference which are of interest for us in this thesis, are divided into two broad classes, namely: wideband interference and narrowband interference. While narrowband interference is briefly discussed in this subsection, wideband multiple access interference is presented in the next subsection. DSSS signals, which includes GNSS radionavigation signals, are known to be immune against narrowband interference to a large extent ([27] Ch. 3, pp. 59-67). In fact, GNSS radionavigation signals can withstand narrowband interference signals whose power levels are several orders of magnitude larger than the power level of the radionavigation signals themselves. Nonetheless, narrowband interference as well as few mitigation algorithms are still implemented in our simulator model. If the reader is interested in a comprehensive discussion about the types of narrowband interference which were implemented in the simulator model, namely: *Continuous Wave Interference* (*CWI*) and *Distance Measuring Equipment* (*DME*), then the reader is encouraged to refer to [56]. However, due to its importance for some forthcoming discussions, CWI is still presented briefly herein. As presented in *Equation 5.26*, CWI is implemented in the simulator model via superimposing the real-valued and discrete-time CWI signal $\mathfrak{i}\,(kT_s)$ upon the pre-CWI signal $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$, which results into a post-CWI signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$. Moreover, as presented also in *Equation 5.26*, $\mathfrak{i}\,(kT_s)$ is itself made of a combination of as many as N_{CWI} different frequency-shifted and amplitude-scaled sinusoidal components.

$$\boldsymbol{r}_{\mathrm{IF+D}}(kT_s) = \boldsymbol{z}_{\mathrm{IF+D}}(kT_s) + \mathfrak{i}\,(kT_s); \qquad 5.26$$

$$\mathfrak{i}\,(kT_s) = \frac{\sqrt{P_z}}{10^{(\mathrm{SIR})_{[\mathrm{dB}]}/20}} \sum_{n=0}^{N_{\mathrm{CWI}}-1} \beta_n \sin(2\pi(f_{\mathrm{IF}} + f_{\mathrm{CWI}} + f_n)kT_s)\,; \quad k = 1, 2, \ldots$$

Where:

- P_z denotes the average power of the pre-CWI signal $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$.
- SIR denotes Signal-to-Interference-power-Ratio which is common for all CWI components, and its value is a user-input parameter supplied directly in $[\mathrm{dB}]$ unit by the simulator model's end-user.
- β_n denotes a scaling coefficient for the n^{th} CWI component.
- f_{CWI} denotes a common frequency offset for all CWI components, and its value is a user-input parameter supplied directly by the simulator model's end-user.
- f_n denotes a frequency offset for the n^{th} CWI component.

Equation 5.26 is applicable to every supported radionavigation signal in the simulator model except Galileo E5 signal. Specifically, in the case of active Galileo-E5a receiver, the energy of the CWI signal $\mathfrak{i}\,(kT_s)$ is distributed around the center frequency of Galileo E5a lower-sideband with respect to f_{IF}. To make it clear, in the case of active Galileo-E5a receiver, f_{IF} of *Equation 5.26* must be replaced with $(f_{\mathrm{IF}} - 15.345 \times 10^6)$. However, although N_{CWI}, β_n, f_n, f_{CWI}, P_z, SIR, $\mathfrak{i}\,(kT_s)$, $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$, and $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$ of *Equation 5.26* are not marked accordingly, they are, nevertheless, always associated with one of the four supported radionavigation signals in the simulator model whenever *Equation 5.26* is itself associated with that particular radionavigation signal (such as the case of *Figure 5.2*). Furthermore, the values of N_{CWI}, β_n, and f_n of *Equation 5.26* were rather arbitrary defined in the simulator model; however, for the record, their values are still presented in *Table 5.1*. To conclude this subsection, in the simulator model, it is up to the end-user whether he/she wants CWI to be activated or not. Consequently, if the end-user chooses to deactivate CWI in any given simulation, then the post-CWI signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$ is basically the same as the pre-CWI signal $\boldsymbol{z}_{\mathrm{IF+D}}(kT_s)$ with no incurred modifications whatsoever.

Table 5.1: A breakdown of the values of CWI parameters N_{CWI}, β_n, and f_n which were defined in the simulator model for the four supported radionavigation signals.

Parameter / n	Galileo E1		Galileo E5		GPS L1		GPS L5	
	β_n	f_n [MHz]	β_n	f_n [MHz]	β_n	f_n [MHz]	β_n	f_n [MHz]
0	1	0.6	1	0.6	1	0.6	1	0.6
1	1	1	1	1	1	1	1	1
2	1	5	0.9	5	0.9	−0.3	0.9	5
3	0.75	−4	0.75	−4	0.75	−1.5	0.75	−4
$4 = N_{\mathrm{CWI}} - 1$	1	0	1	0	1	0	1	0

5.3.4.2 Wideband Multiple Access Interference

The wideband multiple access interference among existing intra-constellation and/or inter-constellation GNSS radionavigation signals, which is termed as MAI-A/R, was verbosely discussed in *Section 4.2*. However, the worst-case scenario of MAI-A/R in the simulator model consists of four distinct radionavigation signals which all share the same IF carrier, so the MAI-A/R effect in the simulator model is clearly an under-representation of real-world circumstances wherein tens of spectrally overlapping radionavigation signals exist simultaneously on the same transmission channel. If needed, this shortcoming of MAI-A/R under-representation could be solved in future simulator model upgrades simply via increasing the number of spectrally overlapping signals until MAI-A/R effect between them starts to bear closer resemblance to the MAI-A/R effect found in the real-world circumstances. Assuming the simulator model's four supported radionavigation signals are all simultaneously active in a given simulation, then the MAI-A/R effect in this case is implemented as in *Equation 5.27*. It is worth mentioning, the MAI-A/R effect is introduced in the simulator model inside the switching unit, which is the large yellow block visible in *Figure 5.1*.

$$\boldsymbol{r}_{\mathrm{IF+D}}(kT_s) = \boldsymbol{r}^{\mathrm{E1}}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{r}^{\mathrm{E5}}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{r}^{\mathrm{C/A}}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{r}^{\mathrm{L5}}_{\mathrm{IF+D}}(kT_s); \quad k = 1, 2, \ldots \qquad 5.27$$

5.4 Receiving Unit

5.4.1 Frontend

Converting an arbitrary real-valued bandpass signal $v_{\mathrm{BP}}(t)$ into a complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(t)$ in the frontend of a practical super-heterodyne receiver is a much more involved process than a mere implementation of *Equation 5.5*, which happens to be followed by a LPF as mentioned earlier. To be more specific, the frontends of some practical super-heterodyne receivers, such as the ones employing the Rockwell Collins's so-called *Miniature Airborne GPS Receiver* (*MAGR*) *architecture*, downconvert a real-valued bandpass signal $v_{\mathrm{BP}}(t)$ into a real-valued IF-centered signal $v_{\mathrm{IF}}(t)$, which is in turn converted to a real-valued lowpass-equivalent composite signal $v_{\mathrm{LP}}(t)$ that is composed of both in-phase and quadrature components, albeit with a small intentional residual frequency offset which is treated in forthcoming receiver processes as though it was a Doppler-shift. This conversion of $v_{\mathrm{BP}}(t)$ into $v_{\mathrm{LP}}(t)$ in the MAGR architecture is carried out totally in the analog-domain through multiple successive stages of filtering and mixing operations. On the other hand, in some other super-heterodyne receivers, such as the ones employing the so-called *NovAtel's GPSCard™ architecture*, a real-valued bandpass signal $v_{\mathrm{BP}}(t)$ is downconverted first to a real-valued IF-centered signal $v_{\mathrm{IF}}(t)$ in the analog-domain, then $v_{\mathrm{IF}}(t)$ is in turn converted to a complex-valued lowpass-equivalent signal $\boldsymbol{v}_{\mathrm{LP}}(kT_s)$ in the digital-domain using a technique known as *IF sampling* (aka, *bandpass sampling, direct conversion*, or *effective intentional aliasing*), that is again with a small intentional residual frequency offset which is treated in forthcoming receiver processes as though it was a Doppler-shift ([27] Ch. 8, pp. 337-348).

In our simulator model, all signals arriving at the receiving units are totally digital, which are also modulated using digital IF carriers as presented in *Equation 5.6*, so there is no need for any sort of frontend to be implemented. Consequently, as shown in *Figure 5.2*, the post-(MAI-A/R) signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$ is passed unaltered from the switching unit to the narrowband interference mitigation module, which is the first module encountered inside every receiving unit. However, the digital IF carrier must be dealt with properly (i.e., must be wiped off) inside both signal acquisition and signal tracking modules, otherwise erroneous results are to be expected.

5.4.2 Narrowband Interference Mitigation

Three different algorithms for narrowband interference mitigation, namely: FFT True Zeroing, Notch Minimum Power, and Pulse Blanking, have been implemented in the simulator model. However, for the sake of conciseness, the details concerning the available algorithms for mitigating narrowband interference are largely omitted herein. Nonetheless, if the reader is interested in a comprehensive discussion about those three mitigation algorithms, then he/she is encouraged to refer to [56]. Regardless of the exact internal mechanism of each interference mitigation algorithm, all of them strive to reach one goal, which is, as much as possible elimination of any existing undesired interference. To be a bit more specific, through various methods, interference mitigation algorithms estimate, as accurately as possible, the undesired interference signal $\mathfrak{i}\,(kT_s)$ to which the desired signal was subjected. Then, as presented in *Equation 5.28*, the estimated interference (denoted as $\hat{\mathfrak{i}}\,(kT_s)$) is eliminated from the post-(MAI-A/R) signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$, which should result into a new signal $\boldsymbol{q}_{\mathrm{IF+D}}(kT_s)$ that is supposedly clean from narrowband interference. Whenever the post-mitigation signal $\boldsymbol{q}_{\mathrm{IF+D}}(kT_s)$ bears closer resemblance to the summation of active pre-CWI signals (e.g., $\boldsymbol{z}^{\mathrm{E1}}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{z}^{\mathrm{E5}}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{z}^{\mathrm{C/A}}_{\mathrm{IF+D}}(kT_s) + \boldsymbol{z}^{\mathrm{L5}}_{\mathrm{IF+D}}(kT_s)$ in case the four supported radionavigation signals were all active), then the better is the underlying interference mitigation algorithm.

$$\boldsymbol{q}_{\mathrm{IF+D}}(kT_s) = \boldsymbol{r}_{\mathrm{IF+D}}(kT_s) - \hat{\mathfrak{i}}\,(kT_s); \quad k = 1, 2, \ldots \qquad 5.28$$

In the simulator model, it is up to the end-user whether he/she wants narrowband interference mitigation to be active or not. Consequently, if the end-user chooses to deactivate narrowband interference mitigation in any given simulation, then the post-mitigation signal $\boldsymbol{q}_{\mathrm{IF+D}}(kT_s)$ is basically the same as the pre-mitigation signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$ with no incurred modifications whatsoever.

5.4.3 Signal Acquisition

While focusing on what was implemented in the simulator model, the topic of GNSS signal acquisition is presented herein in a straightforward fashion. In a nutshell, GNSS signal acquisition is defined as the stage where the presence of a certain satellite's radionavigation signal is either confirmed or dismissed with some margin of error, and in case of confirmation, coarse estimates of both Doppler-shift (denoted as $\widehat{f_{\mathrm{D}}}$) and code-

delay (denoted as $\widehat{\tau_c}$) are made in the process. In other words, GNSS signal acquisition is basically a 3-dimensional search problem, which consists of extracting *Satellite Vehicle Numbers* (*SVN*) of at least four in-view satellites, respective Doppler-shift estimates, and respective code-delay estimates. However, in our simulator model, it is only possible to simulate a single source (i.e., satellite) per activated radionavigation signal, also the receiver is always aware from which satellite(s) the RX-signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$ is generated. Hence, signal acquisition in our simulator model is basically a 2-dimensional search problem. Put simply, signal acquisition module accomplishes its job via generating a 2-D search space with axes representing frequency in one dimension and delay in the other dimension. Then, within the 2-D search space, if the ratio between the global largest power level (i.e., global peak) and the largest power level after suppressing the global peak and its vicinity (i.e., second highest peak) is greater than or equal to a predefined threshold γ, then a signal detection alarm as well as both frequency and delay coordinates of the global peak are all forwarded to the next receiver processing stage, which is signal tracking.

It is to be noted, code-delay τ_c, which is synonymous with Δt of *Subsection 3.2.3*, is not always equivalent to the direct component's propagation delay τ_0 of *Subsection 5.3.2*. To be more specific, if the end-user of the simulator model defines a value for τ_0 that is less than the PRN spreading code repetition-period T_{PRN} (which is usually the case), then τ_c is equivalent to τ_0. Otherwise, τ_c is equivalent to the remainder of the division of τ_0 by T_{PRN}, whereas τ_0 is the dividend and T_{PRN} is the divisor, which is stated more compactly as $\boxed{\tau_c = \tau_0 \bmod T_{\mathrm{PRN}}}$. As presented in *Subsection 5.3.2*, it is to be recalled that the direct component is delayed by a number of samples equivalent to M_0. Therefore, in order to find the estimate of code-delay (denoted as $\widehat{\tau_c}$), then an estimate of the number of samples by which the PRN spreading code is delayed (denoted as $\widehat{M_c}$) must be found first. Similar to the case of code-delay τ_c, if the number of samples by which the PRN spreading code is delayed (denoted as M_c) is less than the total number of samples comprising a T_{PRN}, which is denoted as M_{PRN} and given as $\boxed{(M_{\mathrm{PRN}})_{[\mathrm{sample}]} = \lfloor\, (T_{\mathrm{PRN}})_{[\mathrm{s}]} \times (R_s)_{[\mathrm{sample/s}]} \,\rfloor}$, then M_c is equivalent to the number of samples by which the direct component is delayed M_0. Otherwise, M_c is equivalent to the remainder of the division of M_0 by M_{PRN}, whereas M_0 is the dividend and M_{PRN} is the divisor, which is stated more compactly as $\boxed{M_c = M_0 \bmod M_{\mathrm{PRN}}}$. Once an estimate of M_c

is made in the signal acquisition module, an estimate of τ_c is then computed in a straightforward fashion as $\boxed{(\widehat{\tau_c})_{[\mathrm{s}]} = \left(\widehat{M_c}\right)_{[\mathrm{sample}]} \times (T_s)_{[\mathrm{s/sample}]}}$. However, one should pay close attention that in case any of the reflected multipath components has a power level that is greater than or equal to the direct component's power level (i.e., $(\widetilde{\alpha_n})^2 \geq 1$ for any $n \neq 0$), then the signal acquisition module might be deceived into estimating the code-delay of the reflected component instead of the code-delay of the direct component, thus an estimate of code-delay $\widehat{\tau_c}$ that is far-off from the direct component's code-delay τ_c is produced.

Both axes of the search space are typically of limited resolution; therefore, the frequency axis and the delay axis will be both denoted in the following using a discretized notation, which is f_m and τ_ℓ, respectively. Starting with the description of f_m, searching for the Doppler-shift frequency f_{D} in the signal acquisition module is carried out over a finite number of uniformly spaced frequency steps (aka, frequency bins) which are denoted herein as f_m. In the simulator model, the uncertainty range, which is denoted as f_{limit}, over which f_{D} is searched equally in opposite directions relative to f_{IF}, is not defined by *Equation 5.11*, it is, instead, a user-input parameter. According to the definition of f_m in *Equation 5.29*, the frequency spacing (aka, frequency bin size) between any two successive frequency steps of f_m, which is denoted as Δf_m, is given as $\boxed{\Delta f_m = f_{m+1} - f_m = \xi / T_{\mathrm{coh}}}$. This dimensionless coefficient ξ, which is found in *Equation 5.29* as well as in the aforementioned frequency bin size Δf_m, is a user-input parameter whose value should never exceed 1. According to ([16] Ch. 14, pp. 410-411), the value of coefficient ξ causes attenuation in every correlation's power level in the search space maximally as $\boxed{\left(L_{\max}(\xi)\right)_{[\mathrm{dB}]} = 20\log_{10}\left(\mathrm{sinc}(\xi/2)\right)}$. For example, if $\xi = 2/3$, which is a rule-of-thumb value given in ([13] Ch. 5, pp. 219-231), then due to this resolution of underlying frequency bins, the maximum power loss $L_{\max}(\xi)$ in every correlation performed for the sake of obtaining the search space is expected to be around $-1.65\ [\mathrm{dB}]$. On the other hand, searching for the code-delay τ_c is carried out similarly over a finite number of uniformly spaced delay steps (aka, delay bins in time), which are denoted herein as τ_ℓ. However, signal acquisition modules of the simulator model do not work directly with the delay bins in time τ_ℓ. Instead, signal acquisition modules search for the number of samples by which the PRN spreading code was delayed (denoted by M_c as mentioned earlier) over a finite number of uniformly spaced delay steps in samples (aka, delay bins in samples) which are denoted as M_ℓ and defined in *Equation 5.30*. It is to be recalled,

as shown in a previous formula, once M_c is estimated, an estimate of τ_c could be then computed in a straightforward fashion. Likewise, such as shown in this formula $\boxed{(\tau_\ell)_{[\mathrm{s}]} = (M_\ell)_{[\mathrm{sample}]} \times (T_s)_{[\mathrm{s/sample}]}}$, defining τ_ℓ in terms of M_ℓ is straightforward as well. According to the definition of M_ℓ in *Equation 5.30*, the delay spacing in sample (aka, delay bin size in sample) between any two successive delay steps of M_ℓ, which is denoted as ΔM_ℓ, is given as following: $\boxed{(\Delta M_\ell)_{[\mathrm{sample}]} = (M_{\ell+1} - M_\ell)_{[\mathrm{sample}]} = \left\lfloor (\Delta b)_{[\mathrm{chip}]} \times (R_s/R_c)_{[\mathrm{sample/chip}]} \right\rfloor}$, where Δb is supposed to be a user-input parameter that represents the delay bin size in chips. However, due to acquisition modules' underlying algorithm that is used to generate the search spaces in the simulator model, which is commonly termed *FFT-based parallel code phase search acquisition*, the value of ΔM_ℓ never exceeds $1\ [\mathrm{sample}]$. From the fact that $\Delta M_\ell = 1$ in every acquisition module, we can easily deduce the following: $M_\ell = \ell = 1, 2, 3, \cdots, M_{\mathrm{PRN}}$, which by the way, corresponds to the highest possible resolution in terms of delay bins. That being said, Δb is always implicitly defined as $\boxed{(\Delta b)_{[\mathrm{chip}]} = 1\ [\mathrm{sample}] \times (R_c/R_s)_{[\mathrm{chip/sample}]}}$. Thus, the end-user is never asked for the value of Δb anywhere in the simulator model. Also, the delay bin size in time, which is denoted as $\Delta\tau_\ell$, is always implicitly defined in the simulator model as $\boxed{(\Delta\tau_\ell)_{[\mathrm{s}]} = 1\ [\mathrm{sample}] \times (T_s)_{[\mathrm{s/sample}]}}$. It is worthwhile to mention, each unique pair of (f_m, M_ℓ) constitutes a cell in the search space. Consequently, the total number of cells in any search space inside the simulator model is given as: $\left(2\left\lceil \frac{T_{\mathrm{coh}}}{\xi} f_{\mathrm{limit}} \right\rceil + 1\right) M_{\mathrm{PRN}}$.

$$f_m = \frac{\xi}{T_{\text{coh}}} m; \qquad 5.29$$

$$m = \left\lfloor -\frac{T_{\text{coh}}}{\xi} f_{\text{limit}} \right\rfloor, \ \cdots, \ -1, \ 0, \ 1, \ \cdots, \ \left\lceil \frac{T_{\text{coh}}}{\xi} f_{\text{limit}} \right\rceil$$

$$M_\ell = \left\lfloor \frac{R_s}{R_c} \Delta b \right\rfloor \ell; \qquad \ell = 1, 2, 3, \cdots, \left\lfloor \frac{M_{\text{PRN}}}{\left\lfloor \frac{R_s}{R_c} \Delta b \right\rfloor} \right\rfloor \qquad 5.30$$

Where:

- $\lfloor \cdot \rfloor$ denotes a floor function, which rounds its argument toward $-\infty$.
- $\lceil \cdot \rceil$ denotes a ceiling function, which rounds its argument toward $+\infty$.
- T_{coh} denotes the coherent integration duration, which is an integer multiple of T_{PRN}. In other words, $T_{\text{coh}} = N_{\text{coh}}\, T_{\text{PRN}}$, where N_{coh} is in turn a user-input parameter that is supplied directly as integer number by the end-user of the simulator model.

After describing both discrete axes f_m and ℓ of the search space, we move forward and present the dual channel acquisition of both data channel and pilot channel. It is to be recalled here, in the simulator model, the data channel could be either Galileo E1-B, Galileo E5a-I, GPS L1 C/A, or GPS L5-I. On the other hand, the pilot channel could be either Galileo E1-C, Galileo E5a-Q, or GPS L5-Q. In order to successfully complete the dual channel acquisition, the acquisition module must generate as many as N_{ncoh} successive pairs of both data channel coherent search spaces $\boldsymbol{S}_v^{\text{data}}[f_m, \ell\,]$ and pilot channel coherent search spaces $\boldsymbol{S}_v^{\text{pilot}}[f_m, \ell\,]$, where N_{ncoh} is a user-input parameter that is supplied directly as integer number by the end-user of the simulator model, and the subscript *v* is in turn an integer index used for numbering the pairs. It is to be noted, the probability of correctly detecting a signal is increased to a certain degree whenever the integer number N_{ncoh} is increased. However, increasing the integer number N_{ncoh} has the disadvantage of rendering the signal acquisition a more computationally intensive process. Moreover, due to presence of navigation data and/or secondary code in every channel of Galileo E1, Galileo E5, and GPS L5 radionavigation signals, then from signal acquisition perspective, there is roughly a 50% probability that an ambiguous $180°$ phase shift (aka, polarity reversal) has occurred between any two consecutive repetition-periods of the PRN spreading code belonging to any of the aforementioned channels. Therefore, in order to avoid integrating (or summating in practice) over a polarity reversal, the coherent

integration duration T_{coh} should always be equivalent to T_{PRN} (i.e., $N_{\text{coh}} = 1$) while generating the coherent search spaces for Galileo E1, Galileo E5, or GPS L5. In contrast, GPS L1 C/A signal does not contain secondary code, also the bit-period of its navigation data is $20\ [\text{ms}]$ which is 20 times longer than T_{PRN}; hence, signal acquisition module of GPS L1 C/A radionavigation signal is the only module in the simulator model which is permitted to have $N_{\text{coh}} > 1$. Nonetheless, for the sake of simplicity, signal acquisition of GPS L1 C/A radionavigation signal is always assumed herein to be carried out with $N_{\text{coh}} = 1$. In short, for signal acquisition of any radionavigation signal presented herein, $T_{\text{coh}} = T_{\text{PRN}}$.

One should pay close attention that GPS L1 C/A radionavigation signal does not have pilot channel; hence, the pilot channel coherent search space $\boldsymbol{S}_v^{\text{pilot}}[f_m, \ell\,]$ is non-existent (or zero-valued if necessary) in GPS L1 C/A signal acquisition module. However, after each pair of $\left(\boldsymbol{S}_v^{\text{data}}[f_m, \ell\,],\ \boldsymbol{S}_v^{\text{pilot}}[f_m, \ell\,]\right)$ from the series of N_{ncoh} successive pairs has been generated according to *Equation 5.31*, both $\boldsymbol{S}_v^{\text{data}}[f_m, \ell\,]$ and $\boldsymbol{S}_v^{\text{pilot}}[f_m, \ell\,]$ of each pair are typically combined together into a single search space. In the GNSS literature, there are several different techniques through which the combined search space of both data and pilot channels is generated [57]. The technique that is implemented in our simulator model, which is presented in *Equation 5.32*, is known as *noncoherent channel combining technique*. Accordingly, the noncoherent combined search space of the v^{th} pair is denoted as $S_v^{\text{ncoh}}[f_m, \ell]$. As presented in *Equation 5.33*, in order to generate the overall noncoherent search space $S_{\text{ncoh}}[f_m, \ell]$ in the case $N_{\text{ncoh}} > 1$, then the search spaces found in the series of N_{ncoh} noncoherent combined search spaces $S_v^{\text{ncoh}}[f_m, \ell]$ have to be all algebraically added together. Since each noncoherent combined search space $S_v^{\text{ncoh}}[f_m, \ell]$ is computed over a duration of T_{coh} from the post-mitigation signal $\boldsymbol{q}_{\text{IF+D}}(kT_s)$, the overall noncoherent search space $S_{\text{ncoh}}[f_m, \ell]$ is computed over a duration of T_{ncoh} from $\boldsymbol{q}_{\text{IF+D}}(kT_s)$, where T_{ncoh} is known as the *dwell time* and given as $T_{\text{ncoh}} = N_{\text{ncoh}}\, T_{\text{coh}}$.

$$\boldsymbol{S}_v^{\text{data}}[f_m, \ell\,] = \text{IFFT}_n\{\, \boldsymbol{Q}_v[f_m, n]\ \ \boldsymbol{C}_{\text{data}}^*[n]\,\}; \qquad \textit{5.31}$$

$$\boldsymbol{S}_v^{\text{pilot}}[f_m, \ell] = \text{IFFT}_n\big\{\, \boldsymbol{Q}_v[f_m, n]\ \ \boldsymbol{C}_{\text{pilot}}^*[n]\,\big\};$$

$$\boldsymbol{Q}_v[f_m, n] = \ \text{FFT}_\ell\big\{\, \boldsymbol{q}_{\text{IF+D}}\big((vM_{\text{PRN}} + \ell)T_s\big)\, e^{-j2\pi(f_{\text{IF}} + f_m)\ell T_s}\,\big\};$$

$$\boldsymbol{C}_{\text{data}}[n] = \ \text{FFT}_\ell\{\, c_{\text{data}}(\ell T_s)\,\}; \qquad \boldsymbol{C}_{\text{pilot}}[n] = \ \text{FFT}_\ell\big\{\, c_{\text{pilot}}(\ell T_s)\,\big\}$$

$$S_v^{\mathrm{ncoh}}[f_m, \ell] = \left| \boldsymbol{S}_v^{\mathrm{data}}[f_m, \ell] \right|^2 + \left| \boldsymbol{S}_v^{\mathrm{pilot}}[f_m, \ell] \right|^2 \quad 5.32$$

$$S_{\mathrm{ncoh}}[f_m, \ell] = \sum_{v=v_0}^{N_{\mathrm{ncoh}}+v_0-1} S_v^{\mathrm{ncoh}}[f_m, \ell] \quad 5.33$$

Where:

- $c_{\mathrm{data}}(\ell T_s)$ and $c_{\mathrm{pilot}}(\ell T_s)$ denote local replicas of data channel and pilot channel, respectively. The local replicas themselves consist mainly of respective PRN sequences which are sampled at R_s. However, in the case of Galileo E1-B/C composite signal acquisition, respective subcarriers must be included in the local replicas as well. Not to mention, in the case of GPS L1 C/A signal acquisition, the pilot channel local replica $c_{\mathrm{pilot}}(\ell T_s)$ is non-existent (or zero-valued if necessary).
- $\mathrm{FFT}_\ell\{\cdot\}$ denotes a function which computes the Fast Fourier Transform in the domain of delay bins in samples ℓ. It is to be noted, the frequency-domain axis in the output of FFT_ℓ is denoted with the dummy variable n. Also, the frequency bin f_m must be fixed each time FFT_ℓ is called. Therefore, in order to account for all frequency bins f_m while generating $\boldsymbol{Q}_v[f_m, n]$ in practice, then FFT_ℓ must be called $\left(2\left\lfloor \frac{T_{\mathrm{coh}}}{\xi} f_{\mathrm{limit}} \right\rfloor + 1\right)$ number of times inside a loop.
- $\mathrm{IFFT}_n\{\cdot\}$ denotes a function which computes the Inverse Fast Fourier Transform in the frequency-domain n. Consequently, IFFT_n returns its argument back to the domain of delay bins in samples ℓ. It is to be noted, the frequency bin f_m must be fixed each time IFFT_n is called. Therefore, in order to account for all frequency bins of f_m while generating each of $\boldsymbol{S}_v^{\mathrm{data}}[f_m, \ell\,]$ and $\boldsymbol{S}_v^{\mathrm{pilot}}[f_m, \ell]$ in practice, then IFFT_n must be called $\left(2\left\lfloor \frac{T_{\mathrm{coh}}}{\xi} f_{\mathrm{limit}} \right\rfloor + 1\right)$ number of times inside a loop.
- v_0 denotes a user-input integer parameter which determines in multiples of M_{PRN} how many samples of $\boldsymbol{q}_{\mathrm{IF+D}}(kT_s)$ should be initially skipped before starting the signal acquisition process. To make it clear, each group of as many as M_{PRN} consecutive samples of any signal in the simulator model is referred to as a frame. That being said, v_0 determines how many frames of $\boldsymbol{q}_{\mathrm{IF+D}}(kT_s)$ are skipped initially before signal acquisition module is initiated, and hence v_0 is measured in $[\mathrm{frame}]$.

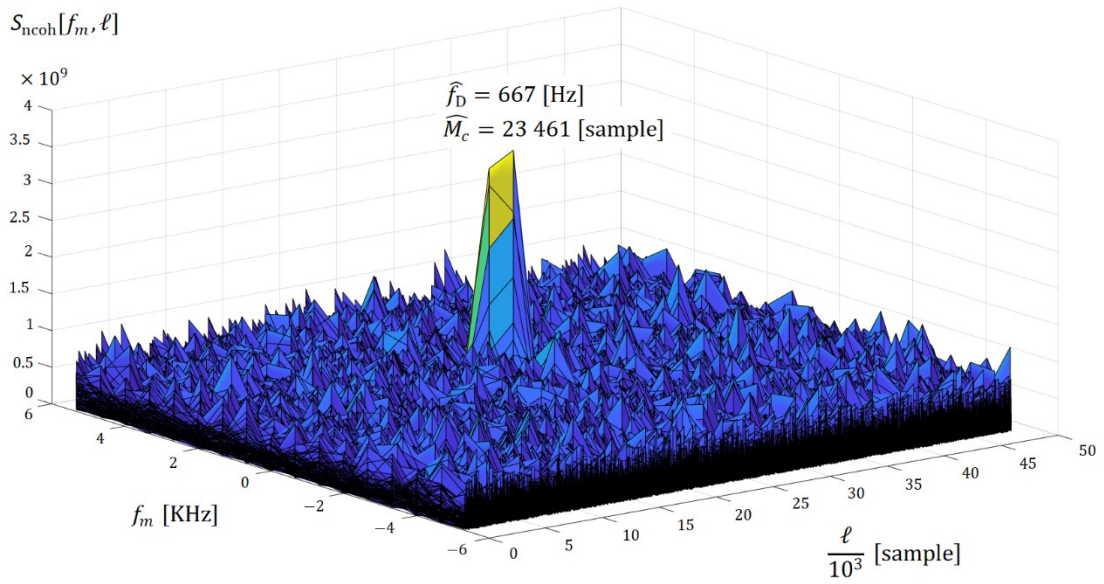

Figure 5.10: Illustration of an overall noncoherent search space of GPS L5-I/Q composite radionavigation signal, which was generated in the simulator model. This search space was obtained with the following set of parameters: $\boldsymbol{x}_{\text{IF}}^{\text{L5}}(kT_s)$ is the only active TX-signal, narrowband interference is deactivated, narrowband interference mitigation is deactivated as well, $R_s = 48 \times 10^6$ [sample/s], $f_{\text{IF}} = 12$ [MHz], $f_{\text{D}} = 998$ [Hz], $N_{\text{mp}} = 1$, $\alpha_0^2 = 0$ [dB], $b_0 = 5\,000$ [chip] $\Rightarrow M_c = 23\,460$ [sample], $C/N_0 = 44$ [dB-Hz], $\Delta f_m = 666.667$ [Hz], $f_{\text{limit}} = 5$ [KHz], $N_{\text{ncoh}} = 1$, and $v_0 = 1$.

Once the milestone of generating the overall noncoherent search space $S_{\text{ncoh}}[f_m, \ell]$ has been reached, such as the one depicted in *Figure 5.10*, the next step is to locate the global peak in $S_{\text{ncoh}}[f_m, \ell]$. Interestingly, once the frequency and delay coordinates of the global peak are extracted, they initially represent both $\widehat{f_{\text{D}}}$ and $\widehat{M_c}$, respectively, which is stated more compactly as: $\max\{ S_{\text{ncoh}}[f_m, \ell] \} = S_{\text{ncoh}}[\widehat{f_{\text{D}}}, \widehat{M_c}]$. However, since the signal under acquisition might be corrupted due to excessive underlying noise, or even absent altogether, $\widehat{f_{\text{D}}}$ and $\widehat{M_c}$ cannot be taken at face value. Although in our simulator model there is no such case as absent signal, the stochasticity of noise in the signal under acquisition is still an issue to be reckoned with. Particularly, in case the underlying noise was overwhelming, the signal under acquisition does not correlate properly with its local replica, even if aligned. Therefore, before judging the validity of $\widehat{f_{\text{D}}}$ and $\widehat{M_c}$, the ratio between the global peak and the second highest peak in $S_{\text{ncoh}}[f_m, \ell]$ must be computed first. This ratio between the two peaks is then compared with the threshold γ, which is a user-input parameter. Depending on the result of the comparison, the signal under acquisition is deemed as either detected or absent/corrupted. Subsequently, the next processing stage, which is signal tracking, is initiated and provided with the values of $\widehat{f_{\text{D}}}$ and $\widehat{M_c}$ only in case the signal was deemed detected in the acquisition stage. Lastly, in

order to locate the second highest peak in $S_{\mathrm{ncoh}}[f_m, \ell]$, the global peak and its vicinity must be suppressed first. More specifically, the vicinity of the global peak which ought to be suppressed is the region in $S_{\mathrm{ncoh}}[f_m, \ell]$ that is bounded by both f_m^{supp} and ℓ_{supp}, whose values are rigorously defined in *Equation 5.34*. As presented also in *Equation 5.34*, $S_{\mathrm{supp}}[f_m, \ell]$ denotes the search space in which the global peak and its vicinity have been suppressed.

5.34

$$S_{\mathrm{supp}}[f_m, \ell] = S_{\mathrm{ncoh}}[f_m, \ell] - S_{\mathrm{ncoh}}\left[f_m^{\mathrm{supp}}, \ell_{\mathrm{supp}}\right];$$

$$f_m^{\mathrm{supp}} = \begin{cases} f_m < \widehat{f_{\mathrm{D}}} + f_{\mathrm{s}}, & \text{when } \widehat{f_{\mathrm{D}}} - f_{\mathrm{s}} < \min\{ f_m \}; \\ f_m > \widehat{f_{\mathrm{D}}} - f_{\mathrm{s}}, & \text{when } \widehat{f_{\mathrm{D}}} + f_{\mathrm{s}} > \max\{ f_m \}; \\ \widehat{f_{\mathrm{D}}} - f_{\mathrm{s}} < f_m < \widehat{f_{\mathrm{D}}} + f_{\mathrm{s}}, & \text{otherwise;} \end{cases}$$

$$\ell_{\mathrm{supp}} = \begin{cases} \left(\ell < \widehat{M_c} + M_{\mathrm{s}}\right) + \left(\ell > \widehat{M_c} - M_{\mathrm{s}} + M_{\mathrm{PRN}}\right), & \text{when } \widehat{M_c} - M_{\mathrm{s}} < 0 \\ \left(\ell > \widehat{M_c} - M_{\mathrm{s}}\right) + \left(\ell < \widehat{M_c} + M_{\mathrm{s}} - M_{\mathrm{PRN}}\right), & \text{when } \widehat{M_c} + M_{\mathrm{s}} > M_{\mathrm{PRN}} \\ \widehat{M_c} - M_{\mathrm{s}} < \ell < \widehat{M_c} + M_{\mathrm{s}}, & \text{otherwise} \end{cases}$$

Where:

- f_{s} denotes a number that determines the range of frequencies which are suppressed, if possible, equally in opposite directions relative to the frequency coordinate of the global peak in $S_{\mathrm{ncoh}}[f_m, \ell]$. It is to be noted, the value of f_{s} is a user-input parameter.
- M_{s} denotes the number of samples which are suppressed equally in opposite directions relative to the delay coordinate in sample of the global peak in $S_{\mathrm{ncoh}}[f_m, \ell]$. It is to be noted, the value *of* M_{s} itself is defined as $(M_{\mathrm{s}})_{[\mathrm{sample}]} = \left\lfloor (b_{\mathrm{s}})_{[\mathrm{chip}]} \times (R_s/R_c)_{[\mathrm{sample/chip}]} \right\rfloor$, where b_{s} in turn denotes a user-input parameter that represents the number of chips which are suppressed equally in opposite directions relative to the global peak's delay coordinate in chip.

5.4.4 Signal Tracking

Tracking of GNSS radionavigation signals is an immensely detailed and complicated topic, both practically and theoretically. Due to the constraints on how many pages an

M.Sc. thesis should maximally be, and due to time shortages as well, most of the technical details of the signal tracking modules which were implemented in the simulator model are omitted herein. In fact, if a complete description of tracking modules were to be included in this M.Sc. thesis, it would have easily added extra 20 to 30 pages to this already long M.Sc. thesis. However, since a great deal of the tracking modules was inherited from earlier versions of the simulator model, the reader can refer to [55, 56] as well as ([13] Ch. 5). In case the reader was keen on learning the inner-workings of the tracking modules, the author strongly recommends a personal inspection of the tracking modules which were implemented inside the simulator model (by the way, the entire simulator model is published with an open-source license).

The following is a quick recapitulation of the main aspects of the tracking modules that were implemented in the simulator model:

- Signal tracking starts executing only after the signal under acquisition has been deemed as detected by the signal acquisition module (i.e., $\max\{ S_{\text{ncoh}}[f_m, \ell] \} \div \max\{ S_{\text{supp}}[f_m, \ell] \} \geq \gamma$), regardless if it is a true detection or a false-alarm.
- For all the supported radionavigation signals in the simulator model, dual-channel tracking of both data and pilot channels is implemented by default, except for GPS L1.
- For better efficiency, the integrate-and-dump accumulators (aka, correlators) as well as the *Numerically Controlled Oscillator*s (*NCO*) necessary for the carrier tracking loop and the code tracking loop are all implemented using C-MEX S-Functions, which are written in the C programming language.
- *The Predetection Integration Time* (*PIT*), which is defined as the coherent integration time for the integrate-and-dump accumulators (aka, correlators), is a user-input parameter.
- Contrary to most practical GNSS receivers, the banks of correlators in the tracking modules are not restricted to contain the Early (E), Prompt (P), and Late (L) replicas only. In fact, it is up to the end-user to decide how many replicas should the bank of correlator contain within any tracking module.
- The phase-spacing (i.e., delay) between the Early (E) and the Late (L) replicas, which is denoted by $\Delta_{\text{E-L}}$ and measured in $[\text{chip}]$ unit, is a user-input parameter. $\Delta_{\text{E-L}}$ is also equivalent to double the phase-spacing between any two successive replicas found in the tracking module's bank of correlators.

- The carrier tracking loop filter is implemented using a second-order *Phase-Locked Loop* (*PLL*) assisted with first-order *Frequency-Locked Loop* (*FLL*). Furthermore, the PLL discrimination is implemented using a two-quadrant inverse tangent Costas discriminator, while the FLL discrimination is implemented using a four-quadrant inverse tangent discriminator.
- The natural radian frequency of the FLL, which is denoted as ω_{0f}, is defined in every tracking module as $\boxed{\omega_{0f} = 4 \times B_{nf}}$, where B_{nf} is in turn the frequency noise bandwidth of the carrier loop filter, which itself is a user-input parameter. On the other hand, the natural radian frequency of the PLL, which is denoted as ω_{0p}, is defined in every tracking module as $\boxed{\omega_{0p} = B_{np}/0.53}$, where B_{np} is in turn the phase noise bandwidth of the carrier loop filter, which itself is also a user-input parameter.
- The natural radian frequency of the *Delay-Locked Loop* (*DLL*), which is denoted herein as ω_{DLL}, is defined in every tracking module as $\boxed{\omega_{\mathrm{DLL}} = 4 \times B_{\mathrm{DLL}}}$, where B_{DLL} is in turn the frequency noise bandwidth of the code loop filter, which itself is a user-input parameter.

5.5 User Interface (UI)

5.5.1 User-Input Parameters

The user-input parameters (aka, options or settings) which are prompted to the end-user as part of the GUI dialog box for gathering user-input parameters, which is shown in *Figure 5.11*, are listed as following:

- Select a viable combination of the four supported GNSS radionavigation signals to be activated at the TX-side.
- For each activated radionavigation signal at TX-side, select a *Satellite Vehicle Number* (*SVN*). However, it is to be noted that one of the shortcomings in the simulator model, is that, it is only possible to simulate a single source (i.e., satellite) per activated signal.
- For each activated radionavigation signal at TX-side, supply the value of the parameter C/N_0 in $[\mathrm{dB\text{-}Hz}]$ unit.
- For each activated radionavigation signal at TX-side, create an ordered list, whose length is N_{mp}, of multipath received powers α_n^2 in $[\mathrm{dB}]$ unit. The value of N_{mp} should never exceed 5.

- For each activated radionavigation signal at TX-side, create an ordered list, whose length is also N_{mp}, of multipath propagation delays b_n in $[\mathrm{chip}]$ unit. N_{mp} should have the same value as its counterpart in the previous step.
- For each activated radionavigation signal at TX-side, supply a value for the Doppler-shift frequency f_{D} with respect to f_{IF} in $[\mathrm{Hz}]$ unit. However, when Galileo E5a receiver is active, the Doppler-shift frequency f_{D} is with respect to the center frequency of Galileo E5a lower-sideband, which is in turn with respect to f_{IF}. Put simply, when Galileo E5a receiver is active, the Doppler-shift frequency f_{D} is with respect to $(f_{\mathrm{IF}} - 15.345 \times 10^6)$.
- Select one of the four available receivers which will be used to process the RX-signal $\boldsymbol{r}_{\mathrm{IF+D}}(kT_s)$, where each receiver is dedicated entirely for processing a distinct radionavigation signal that is supported by the simulator model. It is to be noted, even though E5b upper-sideband continues to exist throughout receiver stages, the Galileo E5a receiver processes only the E5a lower-sideband part from the full-band E5 signal.
- Select type of narrowband interference, which should be either one of the following three options: no interference, *Continuous Wave Interference* (*CWI*), or *Distance Measuring Equipment* (*DME*). It is to be noted, regardless of how many radionavigation signals were activated in the TX-side, interference applies exclusively to the radionavigation signal which is associated with the receiver that was activated in the previous step. Moreover, DME interference is applicable only in case either GPS L5 or Galileo E5a receiver was activated in the previous step.
- If either CWI or DME narrowband interference was selected in the previous step, then supply the value of the *Signal to Interference power Ratio* (SIR) in $[\mathrm{dB}]$ unit.
- If either CWI or DME narrowband interference was selected two steps earlier, then supply the value of the interference frequency offset f_{CWI} with respect to f_{IF} in $[\mathrm{MHz}]$ unit. However, when Galileo E5a receiver is active, the interference frequency offset f_{CWI} is with respect to the center frequency of Galileo E5a lower-sideband, which is in turn with respect to f_{IF}. Put simply, when Galileo E5a receiver is active, the interference frequency offset f_{CWI} is with respect to $(f_{\mathrm{IF}} - 15.345 \times 10^6)$.
- Select type of narrowband interference mitigation, which should be either one of the following four options: no mitigation, FFT True Zeroing, Notch Minimum Power, or Pulse Blanking.

- In case GPS L1 was the receiver activated 5 steps earlier, then supply the value of the coherent integration number N_{coh} for the signal acquisition module, which represents an integer multiplier of T_{PRN} for GPS L1 (which is $1\ [\mathrm{ms}]$). On the other hand, in case either Galileo E1, Galileo E5a, or GPS L5 receiver was activated 5 steps earlier, then the value of coherent integration number N_{coh} is not needed in this case because it is always implicitly defined as 1.
- Supply the value of the noncoherent integration number N_{ncoh} for the signal acquisition module of the activated receiver, which is an integer multiplier of the coherent integration duration T_{coh} that was specified in the previous step.

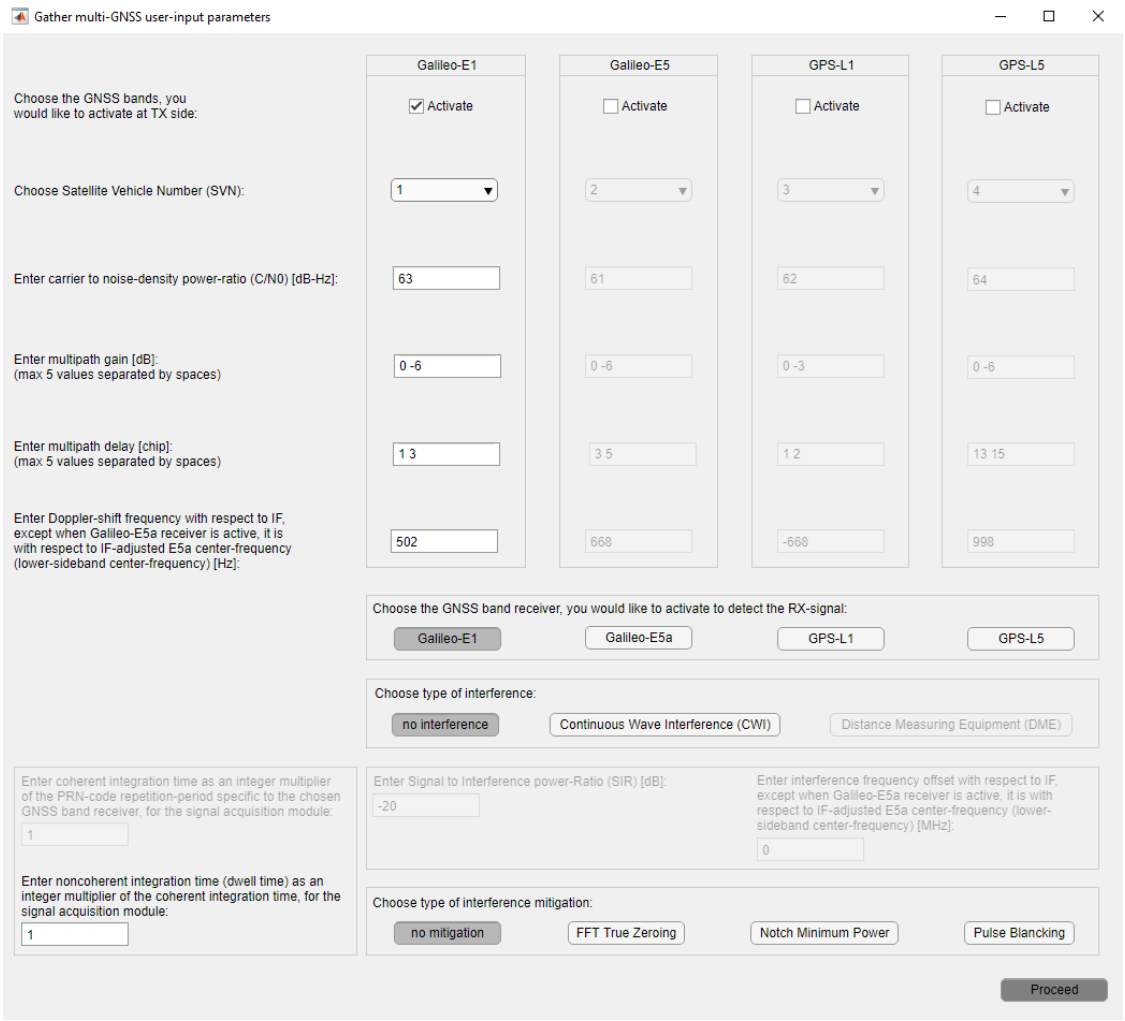

Figure 5.11: An example screenshot of the simulator model's GUI dialog box for gathering user-input parameters.

One of the shortcomings in the simulator model, is that, the GUI dialog box of *Figure 5.11*, whose job is to gather user-input parameters, is not inclusive of all user-input parameters that are available in the simulator model. In fact, several user-input parameters exist in sporadic .m script files which are dispersed across the Simulink® project package. Some of those user-input parameters, which are excluded from the GUI dialog box of *Figure 5.11*, are sometimes difficult to find, and thus the end-user must dig deep into the simulator model if he/she wishes to modify their default values. The following is a list of major user-input parameters which are excluded from the GUI dialog box of *Figure 5.11*:

- The fundamental sampling-rate R_s, which must be configured after considering both f_{IF} and the largest bandwidth among activated radionavigation signals.
- f_{IF}, which is usually assigned the value of $R_s/4$ by default. However, if the end-user wishes, he/she can manually override the default f_{IF} by assigning it any other viable value.
- f_{limit} for each available signal acquisition module, which is defined as the uncertainty range over which f_{D} is searched equally in opposite directions relative to f_{IF}.
- The dimensionless coefficient ξ for each available signal acquisition module, whose value is used for obtaining the frequency bin size Δf_m.
- f_{s} for each available signal acquisition module, which is defined as the range of frequencies that are suppressed, if possible, equally in opposite directions relative to the frequency coordinate of the global peak in $S_{\mathrm{ncoh}}[f_m, \ell]$.
- b_{s} for each available signal acquisition module, which is defined as the number of chips that are suppressed equally in opposite directions relative to the delay chip coordinate of the global peak in $S_{\mathrm{ncoh}}[f_m, \ell]$.
- The threshold γ for each available signal acquisition module, whose value is compared against the ratio between the global peak and the second highest peak in $S_{\mathrm{ncoh}}[f_m, \ell]$.
- The number of bits comprising the holding registers in the NCOs of both code and carrier tracking loops for every signal tracking module, which is denoted as N_{reg}.
- PIT of every signal tracking module, which has to be supplied as an integer multiplier of the repetition-period T_{PRN} for the PRN spreading code specific to the radionavigation signal associated with the active receiver.

- The phase-spacing $\Delta_{\text{E-L}}$ for each available signal tracking module. According to [58, 59], $\boxed{\Delta_{\text{E-L}} \geq 2R_c/B_{\text{BPF}}}$, where R_c is the chipping-rate and B_{BPF} is the double-sided bandwidth for the analog frontend BPF.
- The list of phases in the bank of correlators for each signal tracking module, where each two successive phases are $\Delta_{\text{E-L}}/2$ apart of each other. In order to avoid run-time errors, the length of the list, which is denoted as N_{bank}, must be an odd number, and the list itself must be centered at zero where the zero-valued phase corresponds to the Prompt (P) replica.
- The frequency noise bandwidth of the carrier loop filter B_{nf}, and the phase noise bandwidth of the carrier loop filter B_{np}. Also, the frequency noise bandwidth of the code loop filter B_{DLL}.
- The *Delay-Locked Loop* (*DLL*) type, which could be either one of the following options (options 3, 4, and 5 are applicable for Galileo E1 tracking module):

  1- Early Minus Late (EML).
  2- Single-Stage High Resolution Correlation (HRC).
  3- Dual-Stage HRC.
  4- Single-Stage Multiple Gate Delay (MGD).
  5- Dual-stage MGD.
  6- Dot Product.

5.5.2 Visual End Results

Once a successful termination of any given simulation occurs, the GUI panel of *Figure 5.12*, which is used mainly for visualization of various relevant end results, pops up immediately to the end user. While the left-hand side of the GUI panel shows a summary of major user-input parameters which are associated with the latest simulation, the right-hand side of the GUI panel interactively depicts a handful of end results which are also associated with the latest simulation to the end user. The various relevant end results which are depicted to the end user in the right-hand side of the GUI panel are divided into three groups, which are listed briefly as following:

- First group contains a handful of figures for circular-correlation functions of signals at different stages of the simulation. The end user is prompted to either plot circular-correlation functions which are averaged over all their available frames or plot any single chosen frame of each circular-correlation function.
- Second group contains a handful of PSD figures of signals at different stages of the simulation. Also, if narrowband interference and/or mitigation was activated, then

second group contains time-domain representations as well as few additional PSD figures which are all related to the activated narrowband interference.

- Third group contains few, but important, figures related mainly to acquisition and tracking modules, namely: overall noncoherent search space $S_{\mathrm{ncoh}}[f_m, \ell]$, outputs of the correlators bank with respect to their phases, both instantaneous error and RMSE error in signal tracking.

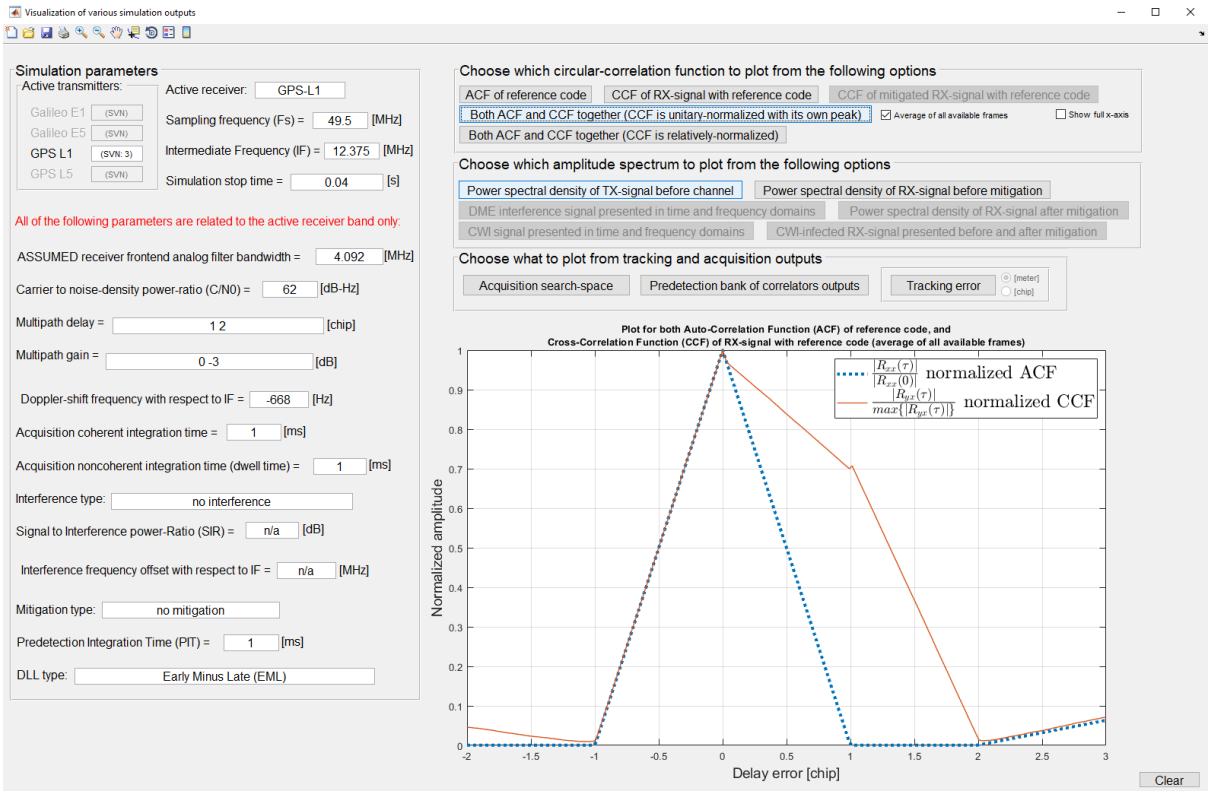

Figure 5.12: An example screenshot of the simulator model's GUI panel for visualization of various relevant end results.

6. PERFORMANCE EVALUATION USING STATISTICAL SIMULATIONS

6.1 Introduction to Statistical Simulation Mode

This chapter focuses mainly on three important parameters, namely: *probability of detection* (denoted as p_{d}), *probability of false-alarm* (denoted as p_{fa}), and tracking *Root-Mean-Square Error* (*RMSE*), all of which are computed against several C/N_0 levels. While the first two parameters are used as metrics for performance evaluation of signal acquisition modules (for more details refer to *Section 6.2*), the third parameter is used as a metric for performance evaluation of signal tracking modules (for more details refer to *Section 6.3*). None of these parameters is obtained through running the simulator model in any sort of a regular simulation mode. In fact, in order to obtain any of those parameters, the end user must perform, what is termed herein as, a statistical simulation. Although some aspects of the difference between the regular simulation mode and the statistical simulation mode were mentioned briefly in the beginning of the previous chapter, the major difference between them is, however, still unmentioned so far. Therefore, driven by its necessity for this chapter, the major difference between regular simulation mode and statistical simulation mode is stated concisely in the following. The regular simulation mode, which was discussed verbosely in the previous chapter, consists of running a single simulation instance (or experiment) inside the simulator model with user-input parameters whose supplied values remain fixed throughout the experiment period. However, before stating in what sense the statistical simulation mode differs from the regular simulation mode, we affirm that everything which was mentioned in the previous chapter is also applicable to the statistical simulation mode, except for the user-input parameters' GUI dialog box and the end results' GUI panel which are both disabled by default in the case of statistical simulation mode. Nonetheless, the user-input parameters which are part of the GUI dialog box are equally important in the statistical simulation mode, and hence they must be supplied manually by the end user if, for any reason, the GUI dialog box cannot be enabled. In the statistical simulation mode, several subordinate independent simulation experiments must be conducted sequentially (or simultaneously if possible) per overall statistical simulation. Furthermore, during each individual subordinate independent simulation experiment, the supplied values of the user-input parameters remain fixed throughout the experiment period. However, when a transition from a

subordinate independent simulation experiment to another occurs, some user-input parameters may change values (e.g., C/N_0).

From the previous paragraph, we conclude that statistical simulation mode is much more involved in terms of computations than regular simulation mode. For example, in order to complete an overall statistical simulation from which Probabilities of Detection p_d for 8 different C/N_0 levels is computed, then as many as 4000 subordinate independent simulation experiments must be conducted in total, that is when 500 subordinate independent simulation experiments are conducted per C/N_0 level. In fact, even when a computer with capable hardware is used, it is not strange for some statistical simulations' processing times T_sim to last up to one week. Therefore, in order to complete the statistical simulations which were needed to obtain the results presented in the continuation of this chapter, a remotely controlled computer-cluster called Narvi, which was provided to the author by the university, was extensively used throughout this M.Sc. thesis. However, since the Narvi computer-cluster itself is a command-line system based on Linux's *slurm workload manager*, such as shown in *Figure 6.1*, the simulator model had to be modified in order to make it compatible with a MATLAB/Simulink® that is running inside a user-interface environment which supports CLI shells exclusively. For example, the modification entailed that the simulator model should be able to refrain from using all GUI features while a statistical simulation is underway. Lastly, without spending much time on the intricacies of both *statistics* and *probability theory*, the following two sections are presented while focusing on what is of interest for this M.Sc. thesis.

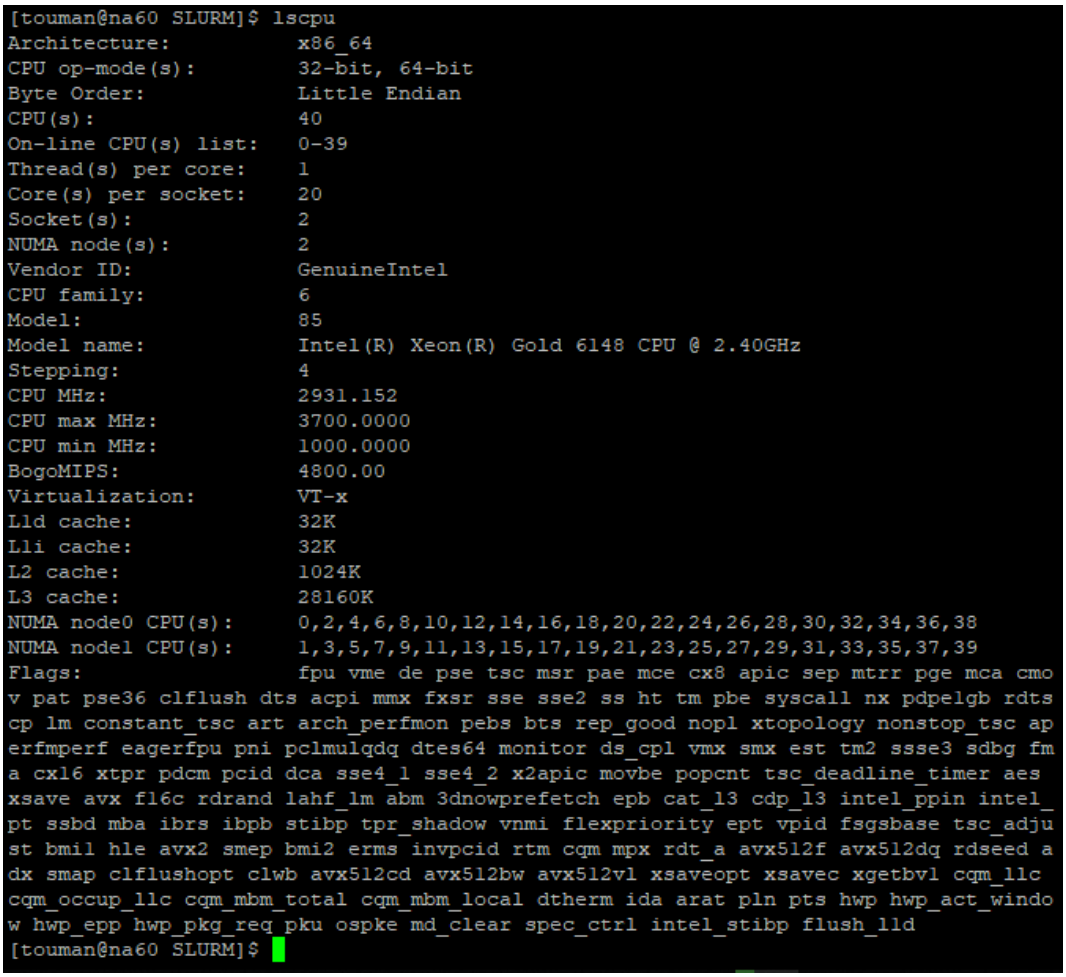

Figure 6.1: A screenshot depicting a summary of CPU architecture information for a node in Narvi computer-cluster called na60. Other nodes' hardware capabilities are not widely different from those of na60, or even exactly matching in many cases.

6.2 Probability of Detection and Probability of False-Alarm in Signal Acquisition

As a starting point for this section, we define a *null hypothesis* (denoted as H_0) and an *alternative hypothesis* (denoted as H_1), which both are given in *Equations 6.1* and *6.2*, respectively. Using *DeMorgan's Theorem* from *Boolean Algebra*, one can easily prove that the null hypothesis H_0 and the alternative hypothesis H_1 given in *Equations 6.1* and *6.2*, respectively, are both complement of each other, and hence testing either one of them is enough to determine the other (such as in the case of *Figure 6.2*). It is to be noted, since presence of the signal under acquisition is always guaranteed in our simulator model, the criterions of signal presence and absence could be dropped from the definitions of H_0 and H_1 in *Equations 6.1* and *6.2* without any conceivable damage to the accuracy of upcoming results. However, the criterions of signal presence and absence

were added to the definitions of H_0 and H_1 in *Equations 6.1* and *6.2* in order to make both equations compatible with the general case which includes, among others, absence of signals under acquisition. As mentioned in *subsection 5.4.3*, the ratio between the global peak and the second highest peak in $S_{\text{ncoh}}[f_m, \ell]$ is, at a certain stage, compared with the threshold γ. Subsequently, the signal under acquisition is deemed detected if the ratio between the two peaks is greater than or equal to γ. In other words, the signal under acquisition is deemed detected if $\max\{ S_{\text{ncoh}}[f_m, \ell] \} \div \max\{ S_{\text{supp}}[f_m, \ell] \} \geq \gamma$. However, there exists a possibility that a signal under acquisition which has been deemed detected by the signal acquisition module is in fact falsely detected due to the stochastic nature of the underlying noise. This possibility is quantified using the Probability of False-Alarm p_{fa}, which is defined as in *Equation 6.2*. On the other hand, probability of detection p_{d}, whose definition is given in *Equation 6.1*, quantifies the likelihood that the signal under acquisition is correctly detected.

$$p_{\text{d}} = P\left(\frac{\max\{ S_{\text{ncoh}}[f_m, \ell] \}}{\max\{ S_{\text{supp}}[f_m, \ell] \}} \geq \gamma \middle| H_0 \right); \qquad 6.1$$

$$H_0 := \text{signal is present } \textbf{and } \left| \widehat{f_{\text{D}}} - f_{\text{D}} \right| \leq \Delta f_m \textbf{ and } |\widehat{M_c} - M_c| \leq M_\varepsilon$$

$$p_{\text{fa}} = P\left(\frac{\max\{ S_{\text{ncoh}}[f_m, \ell] \}}{\max\{ S_{\text{supp}}[f_m, \ell] \}} \geq \gamma \middle| H_1 \right); \qquad 6.2$$

$$H_1 := \text{signal is absent } \textbf{or } \left| \widehat{f_{\text{D}}} - f_{\text{D}} \right| > \Delta f_m \textbf{ or } |\widehat{M_c} - M_c| > M_\varepsilon$$

Where:

- M_ε denotes the maximum tolerated error in the code-delay estimate in sample. It is to be noted, the value of M_ε itself is defined as $(M_\varepsilon)_{[\text{sample}]} = \left| (b_\varepsilon)_{[\text{chip}]} \times (R_s/R_c)_{[\text{sample/chip}]} \right|$, where b_ε in turn denotes the maximum tolerated error in the code-delay estimate in chip. In the simulator model, $b_\varepsilon = 1\ [\text{chip}]$ for signal acquisition modules of Galileo E5, GPS L1, and GPS L5. However, $b_\varepsilon = 0.35\ [\text{chip}]$ exclusively in the case of Galileo E1 signal acquisition module.

Deriving analytical solutions mathematically for both p_{d} and p_{fa} is outside the scope of this M.Sc. thesis. However, for those who are interested, the GNSS literature ([57] for instance) is rich with mathematically-derived analytical solutions for both p_{d} and p_{fa} which

are applicable to the conditions under study. Although an analytical solution for both p_{d} and p_{fa} is omitted herein, a numerical solution for them is still presented in the continuation of this section. From probability theory, we know that the probability of an arbitrary random event $\mathcal{A}$, which is denoted as $P(\mathcal{A})$, has an alternative non-classical definition, which is based on the *relative frequency approach*. To be more specific, "if the experiment is performed N_{exp} times and the event $\mathcal{A}$ occurs $n_{\mathcal{A}}$ times, then, with a high degree of certainty, the relative frequency $n_{\mathcal{A}}/N_{\mathrm{exp}}$ of the occurrence of $\mathcal{A}$ is close to $P(\mathcal{A})$ (which is stated more compactly as $P(\mathcal{A}) \approx n_{\mathcal{A}}/N_{\mathrm{exp}}$), provided that N_{exp} is sufficiently large" ([33] Ch. 1, pp. 5-11). It is to be noted, estimating $P(\mathcal{A})$ via the relative frequency approach which happens to be implemented in the form of a computational algorithm is a key part of the so-called *Monte Carlo Method*. Therefore, in order to estimate both p_{d} and p_{fa} for the various scenarios needed to elaborate this section, many Monte Carlo simulations were carried out in accordance with the custom-made computational algorithm that is illustrated as a flowchart in *Figure 6.2*. It is to be noted, the computational algorithm of *Figure 6.2* constitutes a significant portion (roughly 50%) of the simulator model's statistical simulation mode,

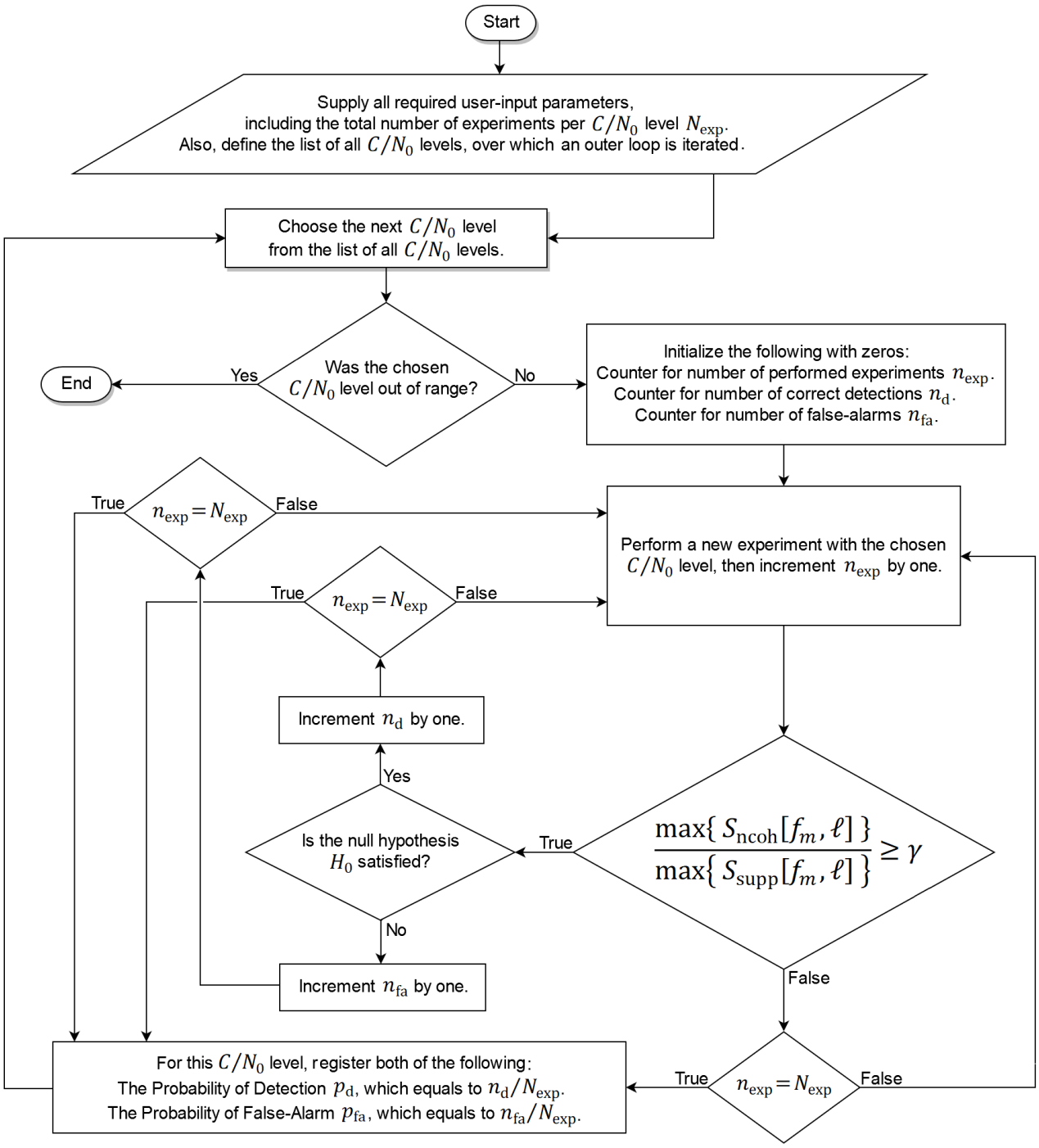

Figure 6.2: A flowchart which represents the computational algorithm that is designed specifically for computing estimates of both p_{d} and p_{fa}.

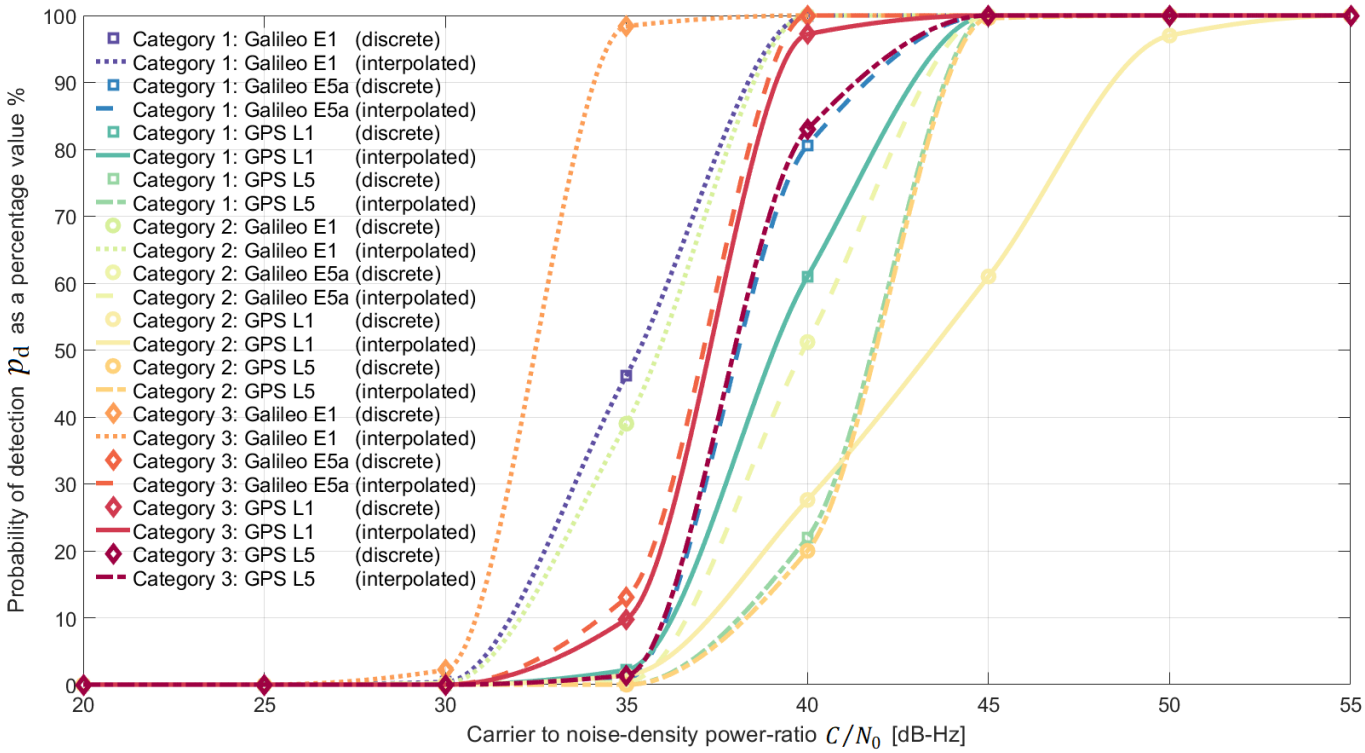

Figure 6.3: An illustration of p_d values against 8 different C/N_0 levels for the four supported radionavigation signals. The statistical simulations are grouped into 3 different categories whose parameters are given in *Table 6.1*. It is to be noted, each discrete data point on the figure was obtained after performing 500 subordinate independent simulation experiments (i.e., $N_{exp} = 500$) consecutively. Also, all curves were obtained using Piecewise Cubic Hermite Interpolating Polynomial (PCHIP).

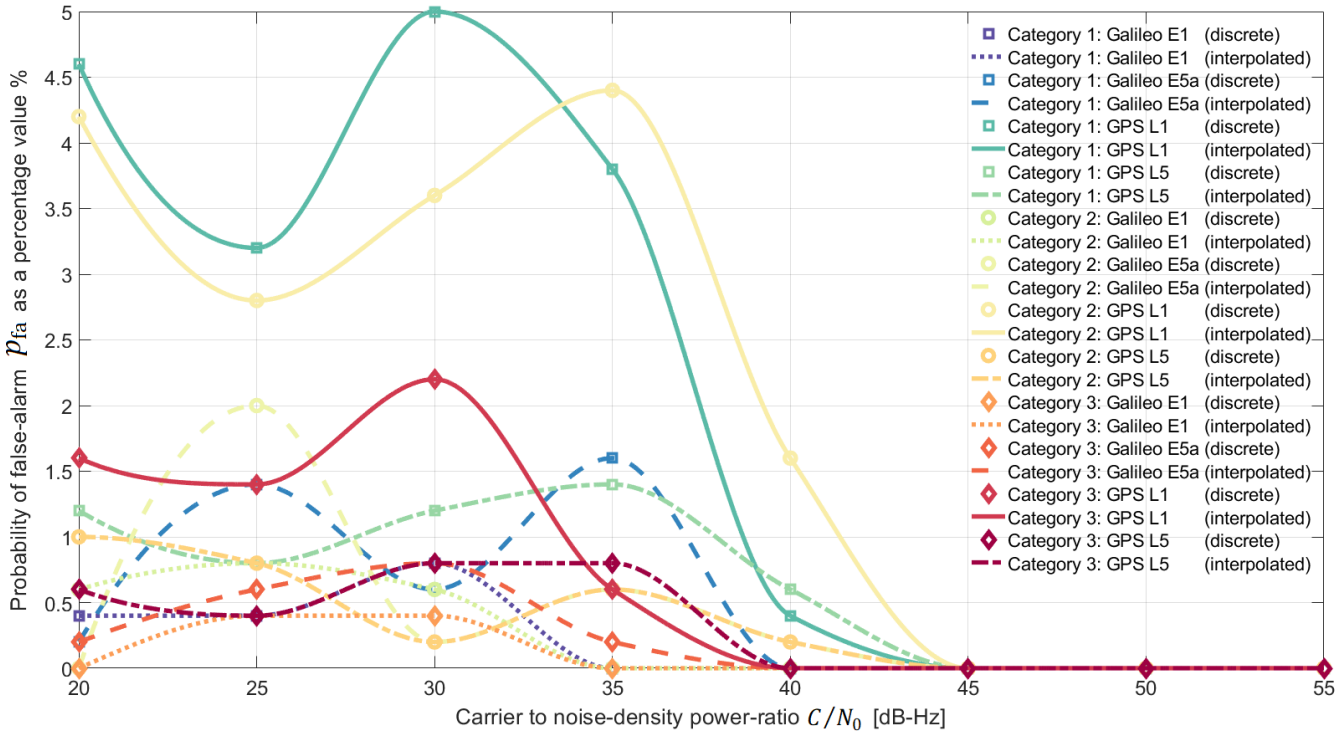

Figure 6.4: An illustration of p_{fa} values against 8 different C/N_0 levels for the four supported radionavigation signals. The statistical simulations are grouped into 3 different categories whose parameters are given in *Table 6.1*. It is to be noted, each discrete data point on the figure was obtained after performing 500 subordinate independent simulation experiments (i.e., $N_{\mathrm{exp}} = 500$) consecutively. Also, all curves were obtained using Piecewise Cubic Hermite Interpolating Polynomial (PCHIP).

Table 6.1: A breakdown of the user-input parameters which are associated with the statistical simulations of both *Figure 6.3* and *Figure 6.4*.

Parameter \ Module		Galileo E1 Acquisition Module	Galileo E5a Acquisition Module	GPS L1 Acquisition Module	GPS L5 Acquisition Module
Common among all categories	Active TX-signals	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{E1}}(kT_s)$	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{E5}}(kT_s)$	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{L1}}(kT_s)$	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{L5}}(kT_s)$
	SVN	15	9	6	15
	R_s [sample/s]	49.5×10^6	104×10^6	8.5×10^6	48×10^6
	f_{IF} [MHz]	12.375	26	2.125	12
	f_{D} [Hz]	375			
	C/N_0 [dB-Hz]	Varies from 20 to 55 in steps of 5.			
	B_{BPF} [MHz]	24.552	24	4.092	24
	Mitigation	Inactive			
	N_{mp}	1			
	α_0^2 [dB]	0			
	b_0 [chip]	1	5	1	5
	N_{coh}	1			
	Δf_m [Hz]	166.667	666.667		
	f_{limit} [KHz]	5			
	v_0 [frame]	1			
	f_{s} [Hz]	1000			
	b_{s} [chip]	3			
	γ	1.3			
Specific for category 1	CWI	Inactive			
	N_{ncoh}	1			
	T_{sim} [hour]	42.48	9.87	2.32	5.10
Specific for category 2	CWI	Active			
	SIR [dB]	-20			
	f_{CWI} [MHz]	0.5			
	N_{ncoh}	1			
	T_{sim} [hour]	42.09	9.84	2.32	5.14
Specific for category 3	CWI	Inactive			
	N_{ncoh}	2			
	T_{sim} [hour]	75.56	14.19	2.47	5.75

6.3 Root-Mean-Square Error (RMSE) in Signal Tracking

Correlators in signal tracking modules repeatedly accumulate PIT duration worth of samples from the signal under tracking. In fact, a signal tracking module produces a new signal delay estimate each time a PIT duration has passed. It is worth mentioning that the simulator model's script which is responsible for computing tracing RMSE in the statistical simulation mode registers all of those signal delay estimates, and at the end of the simulation, computes tracking RMSE from those registered signal delay estimates exactly as given in *Equation 6.3*.

$$(\text{tracking RMSE})_{[\text{m}]} = T_c \cdot c \cdot \sqrt{\frac{1}{N_\sigma}\sum_{n=0}^{N_\sigma - 1}(\sigma - \widehat{\sigma_n})^2} \qquad 6.3$$

Where:

- σ denotes the true signal delay against which estimated signal delay values are compared.
- $\widehat{\sigma_n}$ denotes the n^{th} estimated signal delay value, where each σ_n is computed over a time period equivalent to PIT from the signal under tracking.
- N_σ denotes the total number of estimated signal delay values $\widehat{\sigma_n}$ over which the tracking RMSE is evaluated. For example, if the tracking RMSE is to be evaluated over a 10 [s] worth of the signal under tracking, where PIT is 4 [ms], then $N_\sigma = 2500$ in this case.

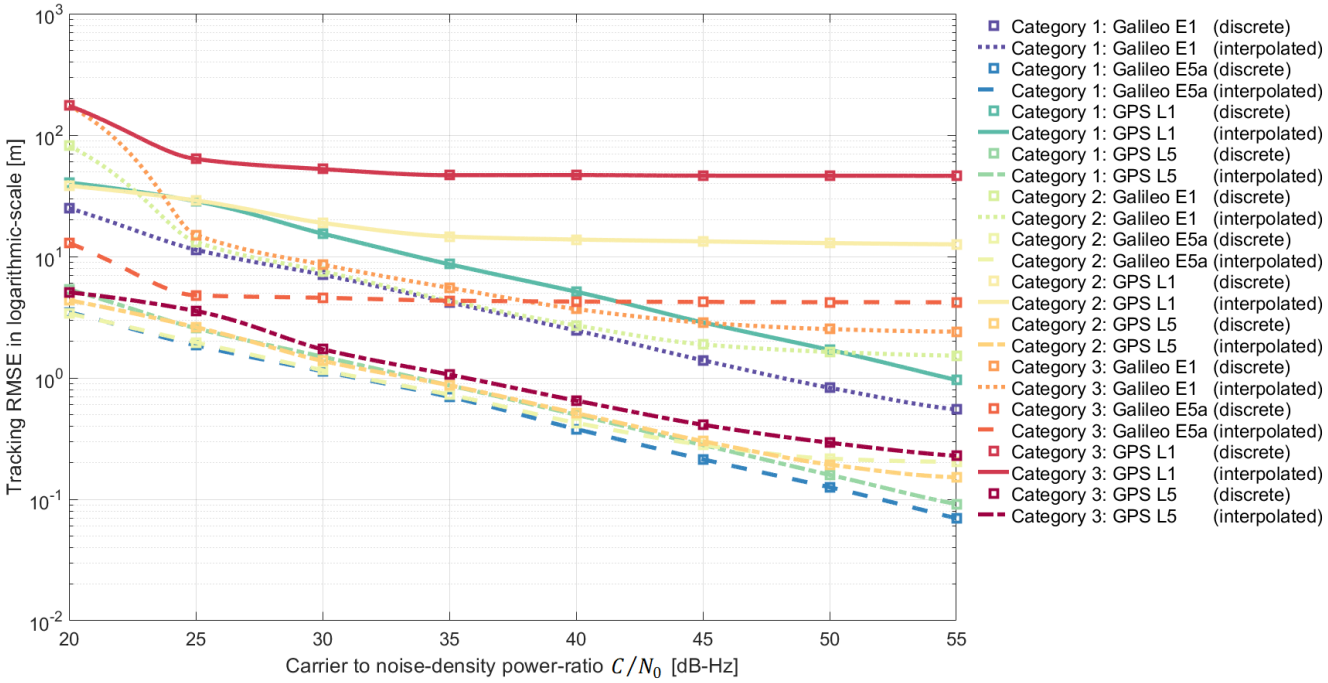

Figure 6.5: An illustration of tracking RMSE values in meter unit against 8 different C/N_0 levels for the four supported radionavigation signals. The statistical simulations are grouped into 3 different categories whose parameters are given in *Table 6.2*. It is to be noted, each discrete data point on the figure was obtained after running the tracking module for exactly 10 seconds of execution time. Also, all curves were obtained using Piecewise Cubic Hermite Interpolating Polynomial (PCHIP).

Table 6.2: A breakdown of the user-input parameters which are associated with the statistical simulations of *Figure 6.5*.

Parameter \ Module		Galileo E1 Tracking Module	Galileo E5a Tracking Module	GPS L1 Tracking Module	GPS L5 Tracking Module
Common among all categories	Active TX-signals	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{E1}}(kT_s)$	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{E5}}(kT_s)$	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{L1}}(kT_s)$	$\boldsymbol{x}_{\mathrm{IF+D}}^{\mathrm{L5}}(kT_s)$
	SVN	10	5	16	21
	R_s [sample/s]	49.5×10^6	104×10^6	8.5×10^6	48.5×10^6
	f_{IF} [MHz]	12.375	26	2.125	12.125
	f_{D} [Hz]	502	668	-335	998
	C/N_0 [dB-Hz]	Varies from 20 to 55 in steps of 5.			
	B_{BPF} [MHz]	24.552	24	4.092	24
	Mitigation	Inactive			
	N_{reg} [bit]	32			
	PIT [ms]	4	1		
	$\Delta_{\mathrm{E\text{-}L}}$ [chip]	0.16	0.5		
	N_{bank}	51	17		
	B_{nf} [Hz]	0.01	2		
	B_{np} [Hz]	10	15		
	B_{DLL} [Hz]	4	8		
	DLL type	normalized Early Minus Late (EML)			
Specific for category 1	CWI	Inactive			
	N_{mp}	1			
	α_0^2 [dB]	0			
	b_0 [chip]	0			
	T_{sim} [hour]	19.60	68.12	3.00	21.12
Specific for category 2	CWI	Active			
	SIR [dB]	-20			
	f_{CWI} [MHz]	0			
	N_{mp}	1			
	α_0^2 [dB]	0			
	b_0 [chip]	0			
	T_{sim} [hour]	22.36	66.16	3.16	20.19
Specific for category 3	CWI	Inactive			
	N_{mp}	2			
	α_0^2 [dB]	$[0, -3]$			
	b_0 [chip]	$[0, 1]$			
	T_{sim} [hour]	22.12	65.96	3.02	20.51

7. CONCLUSIONS

It was substantiated that existence of interoperability among different GNSS constellations is the key for achieving multi-constellation positioning. Particularly, it is an intricate task, if not impossible, to design a receiving equipment capable of multi-constellation positioning without interoperability between the involved constellations. It was also shown that due to the underlying DSSS nature of all existing GNSS radionavigation signals, RFC is maintained to a large extent. However, even with the DS-CDMA scheme in place, RFC is still prone to degradations whenever wideband interference is increased, which consequently puts RFC in a collision course with interoperability since increasing wideband interference is a side effect of interoperability. Therefore, in order to achieve multi-constellation positioning, interoperability must be maintained while ensuring that RFC is kept at a tolerable level, which is the case with both the legacy GPS L1 and the modern Galileo E1 radionavigation signals.

A huge part of this M.Sc. thesis was devoted for an extensive description of the inner-workings of the simulator model for IF-level signal processing of Galileo E1, Galileo E5, GPS L1, and GPS L5 radionavigation signals, including effects of multipath and interference, using MATLAB/Simulink®, which was developed at the former Tampere University of Technology (and now known as Tampere University, Hervanta Campus).

Finally, this M.Sc. thesis was concluded with a performance evaluation of the simulator model in question using statistical simulations. In particular, a graphical presentation is given for probability of detection p_{d}, probability of false-alarm p_{fa}, and tracking RMSE, all of which were evaluated in three different simulation scenarios for each of Galileo E1, Galileo E5, GPS L1, and GPS L5 radionavigation signals.

REFERENCES

[1] R. Lollock, T. Powell, T. Stansell, The Interchangeability Problem: Signals, Coordinate Frames, and Time, in: L. Davis, P. Enge, G. Gao (ed.), Global Navigation Satellite Systems: Report of a Joint Workshop of the National Academy of Engineering and the Chinese Academy of Engineering, The National Academies Press, Washington, DC, 2012, pp. 75-82.

[2] S. K. Moore, Super-Accurate GPS Coming to Smartphones in 2018 [News], IEEE Spectrum, Vol. 54, Iss. 11, 2017, pp. 10-11.

[3] BCM4775X GNSS Receiver with Integrated Sensor Hub (Product Brief), 2017, https://www.broadcom.com/products/wireless/gnss-gps-socs/bcm47755.

[4] Dual-Frequency Smartphone Arrives, GPS World, Vol. 29, Iss. 7, 2018, pp. 40.

[5] U. Robustelli, V. Baiocchi, G. Pugliano, Assessment of Dual Frequency GNSS Observations from a Xiaomi Mi 8 Android Smartphone and Positioning Performance Analysis, Electronics, Vol. 8, Iss. 1 (91), 2019, https://doi.org/10.3390/electronics8010091.

[6] P.D. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, 2nd ed. Artech House, 2013, .

[7] R. Langley, Why Is the GPS Signal So Complex, GPS World, Vol. 1, Iss. 3, 1990, pp. 56-59.

[8] P. Misra, P. Enge, Global Positioning System: Signals, Measurements, and Performance, 2nd ed. Ganga-Jamuna Press, 2006, .

[9] B.W. Parkinson, S.T. Powers, The Origins of GPS: and the Pioneers Who Launched the System, GPS World, Vol. 21, Iss. 5, 2010, pp. 30-41.

[10] Air Commodore Norman Bonnor, A Brief History of Global Navigation Satellite Systems, The Journal of Navigation, Vol. 65, Iss. 1, 2012, pp. 1-14. Available (accessed Copyright © The Royal Institute of Navigation 2011;): .

[11] Pace Scott, Frost Gerald P., Lachow Irving, Frelinger David R., Fossum Donna, Wassem Don, and Pinto Monica M., The Global Positioning System: Assessing National Policies, RAND Corporation, Santa Monica, California, 1995, Available: https://www.rand.org/pubs/monograph_reports/MR614.html.

[12] H. Bray, You Are Here: From the Compass to GPS, the History and Future of How We Find Ourselves, Basic Books, New York, 2014, .

[13] E. Kaplan, C. Hegarty, Understanding GPS: Principles and Applications, 2nd ed. Artech House, 2005, .

[14] Y. Georgiadou, K. Doucet, The Issue of Selective Availability, GPS World, 1990, pp. 53–56. http://gauss.gge.unb.ca/gpsworld/EarlyInnovationColumns/Innov.1990.09-10.pdf.

[15] B. Hofmann-Wellenhof, H. Lichtenegger, E. Wasle, GNSS — Global Navigation Satellite Systems; GPS, GLONASS, Galileo, and more, Springer Vienna, Vienna, 2008, .

[16] P. Teunissen, O. Montenbruck, Springer Handbook of Global Navigation Satellite Systems, Springer International Publishing, 2017, .

[17] M. Matosevic, Z. Salcic, S. Berber, A Comparison of Accuracy Using a GPS and a Low-Cost DGPS, IEEE Transactions on Instrumentation and Measurement, Vol. 55, Iss. 5, 2006, pp. 1677-1683.

[18] D. Massé, GPS/GNSS IC Market on Course to Break $2 B in 2012, Microwave Journal, Vol. 56, Iss. 1, 2013, pp. 61.

[19] J. Raasakka, H. Hurskainen, J. Nurmi, GNSS Baseband Processing in a Multi-Core Platform, 2011 International Conference on Localization and GNSS (ICL-GNSS), pp. 42-46.

[20] U. Roßbach, Positioning and Navigation Using the Russian Satellite System GLONASS - Ph.D. thesis, Universität der Bundeswehr München, 2000, .

[21] P. Ferrão, J. Sanguino, A. Rodrigues, Positioning Using GPS and GLONASS Observations, Technical University of Lisbon, Portugal, 2013, .

[22] R. Langley, GLONASS — Past, Present and Future; An Alternative and Complement to GPS, GPS World, Vol. 28, Iss. 11, 2017, pp. 44. https://www.gpsworld.com/innovation-glonass-past-present-and-future/.

[23] China's BeiDou Navigation Satellite System, Foreign Languages Press Co. Ltd, Beijing, China, 2016, .

[24] J. Nurmi, E. Lohan, S. Sand, H. Hurskainen, GALILEO Positioning Technology, Springer, Netherlands, Dordrecht, 2015, .

[25] S. Bancroft, An Algebraic Solution of the GPS Equations, IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-21, Iss. 1, 1985, pp. 56-59.

[26] J. Nurmi, E. Lohan, H. Wymeersch, G. Seco-Granados, O. Nykänen, Multi-Technology Positioning, Springer International Publishing, 2017, .

[27] B.W. Parkinson, J.J. Spilker Jr., P. Axelrad, P. Enge, Global Positioning System, Volume 1 - Theory and Applications, American Institute of Aeronautics and Astronautics, Inc., 370 L'Enfant Promenade, SW, Washington, DC 20024-2518, 1996, .

[28] M. Cokrlic, K. Wezka, R. Galas, M. M. Hoque, N. Jakowski, V. Wilken, J. Vukovic, Performance Comparison of Selected Ionospheric Models for GNSS Positioning, 2018 International Symposium ELMAR, pp. 173-178.

[29] F. Alizadeh-Shabdiz, M. Heidari, Determining A Dilution of Precision Metric Using Two or Three GPS Satellites, US 2011/0080318 A1, 12/572,516, .

[30] K. Borre, D.M. Akos, N. Bertelsen, P. Rinder, S.H. Jensen, A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach, Birkhäuser Boston, 2007, .

[31] J.R. Barry, E.A. Lee, D.G. Messerschmitt, Digital Communication, 3rd ed. Springer US, 2003, .

[32] European GNSS (Galileo) Open Service Signal-In-Space Interface Control Document; OS SIS ICD Issue 1.3, The European Commission, 2016, .

[33] A. Papoulis, S. Unnikrishna Pillai, Probability, Random Variables, and Stochastic Processes, International Edition, 4th ed. McGraw-Hill Education, New York, 2002, .

[34] IRN-IS-200H-003, Navstar GPS Space Segment/Navigation User Segment Interfaces, Global Positioning Systems Directorate, California, 2015, .

[35] R. Gold, Optimal Binary Sequences for Spread Spectrum Multiplexing (Corresp.), IEEE Transactions on Information Theory, Vol. 13, Iss. 4, 1967, pp. 619-621.

[36] J. Ávila-Rodríguez, On Generalized Signal Waveforms for Satellite Navigation - Ph.D. thesis, Universität der Bundeswehr München, 2008, .

[37] J. Avila-Rodriguez, G.W. Hein, S. Wallner, J. Issler, L. Ries, L. Lestarquit, A. de Latour, J. Godet, F. Bastide, T. Pratt, J. Owen, The MBOC Modulation: The Final Touch to the Galileo Frequency and Signal Plan, InsideGNSS, Vol. 2, Iss. 5, 2007, pp. 43-58. www.insidegnss.com/auto/1007-AvilaRodriguez-final-hires2.pdf.

[38] J.W. Betz, The Offset Carrier Modulation for GPS Modernization, Proceedings of the 1999 National Technical Meeting of The Institute of Navigation, 1999, San Diego, CA, pp. 639-648.

[39] J.W. Betz, Binary Offset Carrier Modulations for Radionavigation, Navigation, Vol. 48, Iss. 4, 2001, pp. 227-246. Available (accessed doi: 10.1002/j.2161-4296.2001.tb00247.x; 05): https://doi.org/10.1002/j.2161-4296.2001.tb00247.x.

[40] E. Rebeyrol, Optimisation des signaux et de la charge utile Galileo - Ph.D thesis, Télécom ParisTech, 2007, .

[41] A. Burian, E. Lohan, M. Renfors, Oversampling Limits for Binary Offset Carrier Modulation for the Acquisition of Galileo Signals, Proceedings of Nordic Radio Symposium and Finnish Wireless Communications Workshop (NRS/FWCW), Oulu, Finland, .

[42] A.F. Molisch, Wireless Communications, 2nd ed. Wiley, Chichester, 2011, .

[43] T. He, Z. Ma, Proposed OFDM Modulation for Future Generations of GNSS Signal System, Journal of Navigation, Vol. 69, Iss. 5, 2016, pp. 971-990.

[44] T. Paakki, Next Generation Multi-System Multi-Frequency GNSS Receivers - Ph.D. thesis, Tampere University of Technology, 2017, .

[45] E. P. Glennon, A. G. Dempster, Delayed PIC for Postcorrelation Mitigation of Continuous Wave and Multiple Access Interference in GPS Receivers, IEEE Transactions on Aerospace and Electronic Systems, Vol. 47, Iss. 4, 2011, pp. 2544-2557.

[46] J. Querol, A. Alonso-Arroyo, R. Onrubia, D. Pascual, H. Park, A. Camps, SNR Degradation in GNSS-R Measurements Under the Effects of Radio-Frequency Interference, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, Vol. 9, Iss. 10, 2016, pp. 4865-4878.

[47] F. Soualle, T. Burger, Radio Frequency Compatibility Criterion for Code Tracking Performance, Institute of Navigation (ION), 2007, pp. 1201-1210. Available (accessed Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation): .

[48] W. Liu, X. Zhan, L. Liu, M. Niu, GNSS RF Compatibility Assessment: Interference among GPS, Galileo, and Compass, GPS World, Vol. 21, Iss. 12, 2010, pp. 32-40. Available (accessed Questex Publications): https://www.gpsworld.com/gnss-rf-compatibility-assessment-10837/.

[49] W. Liu, C. R. Zhai, X. Q. Zhan, Y. H. Zhang, Assessment and analysis of radio frequency compatibility among several global navigation satellite systems, IET Radar, Sonar & Navigation, Vol. 5, Iss. 2, 2011, pp. 128-136.

[50] H. Günter, GNSS Interoperability: Achieving a Global System of Systems or "Does Everything Have to Be the Same"? InsideGNSS, Vol. 1, Iss. 1, 2006, pp. 57-60.

[51] P. Kovár, P. Kacmarik, F. Vejrarka, Low Complex Interoperable GNSS Signal Processor and its Performance, IEEE/ION Position, Location and Navigation Symposium, pp. 947-951.

[52] G. Gibbon, GNSS Interoperability; Not So Easy, After All, InsideGNSS, Vol. 6, Iss. 1, 2011, pp. 28-31.

[53] C. E. Shannon, Communication in the Presence of Noise, Proceedings of the IRE, Vol. 37, Iss. 1, 1949, pp. 10-21.

[54] B.A. Siddiqui, Simulink-Based Acquisition Unit for Galileo E1 CBOC Modulated Signals, Tampere University of Technology, 2010, 78 p.

[55] P. Subedi, Software Simulator and Signal Analysis for Galileo E5 band Signals, Tampere University of Technology, 2014, 63 p.

[56] D. Alonso, Narrowband Interference Rejection Studies for Galileo Signals via Simulink, Tampere University of Technology, 2015, 67 p.

[57] D. Borio, C. O'Driscoll, G. Lachapelle, Coherent, Noncoherent, and Differentially Coherent Combining Techniques for Acquisition of New Composite GNSS Signals, IEEE Transactions on Aerospace and Electronic Systems, Vol. 45, Iss. 3, 2009, pp. 1227-1240.

[58] M. Bhuiyan, K. Ullah, E. Lohan, Quantization Effects on Correlation Losses for Galileo E1 Signal Acquisition and Tracking, Proceedings of SPAMEC, 2011, EURASIP Journal on Advances in Signal Processing, Club-Napoca, Romania, pp. 9-12.

[59] A.J. Van Dierendonck, P. Fenton, T. Ford, Theory and Performance of Narrow Correlator Spacing in a GPS Receiver, NAVIGATION, Journal of The Institute of Navigation, Vol. 39, Iss. 3, 1992, pp. 265-284. https://www.ion.org/publications/abstract.cfm?articleID=100183.